MobilityFirst Future Internet Architecture Project

Ivan Seskar, Kiran Nagaraja, Sam Nelson and Dipankar Raychaudhuri
WINLAB, Rutgers University
Technology Centre of NJ, 671 Route 1
North Brunswick, NJ 08902
seskar@winlab.rutgers.edu

***Abstract* —This short paper presents an overview of the MobilityFirst network architecture, which is a clean-slate project being conducted as part of the NSF Future Internet Architecture (FIA) program. The proposed architecture is intended to directly address the challenges of wireless access and mobility at scale, while also providing new multicast, anycast, multi-path and context-aware services needed for emerging mobile Internet application scenarios. Key protocol components of the proposed architecture are: (a) separation of naming from addressing; (b) public key based self-certifying names (called globally unique identifiers or GUIDs) for network-attached objects; (c) global name resolution service (GNRS) for dynamic name-to-address binding; (d) delay-tolerant and storage-aware routing (GSTAR) capable of dealing with wireless link quality fluctuations and disconnections; (e) hop-by-hop transport of large protocol data units ; and (f) location or context-aware services. The basic operations of a MobilityFirst router are outlined and a simple example of how the protocol supports dual-homing and multi-path is given. This is followed by a discussion of ongoing proof-of-concept prototyping and experimental evaluation efforts for the MobilityFirst protocol stack. In conclusion, a brief description of an ongoing multi-site experimental deployment of the MobilityFirst protocol stack on the GENI testbed is provided.**

Keywords- Future Internet architecture, mobile networks, name resolution, storage-aware routing, GENI prototyping.

I. INTRODUCTION

The MobilityFirst project is founded on the premise that the Internet is approaching an historic inflection point, with mobile platforms and applications poised to replace the fixed-host/server model that has dominated the Internet since its inception. This predictable, yet fundamental, shift presents a unique opportunity to design a next generation Internet in which mobile devices, and applications, and the consequent changes in service, trustworthiness, and management are primary drivers of a new architecture. The major design goals of our proposed architecture are: mobility as the norm with dynamic host and network mobility at scale; robustness with respect to intrinsic properties of wireless medium; trustworthiness in the form of enhanced security and privacy for both mobile networks and wired infrastructure; usability features such as support for context-aware pervasive mobile services, evolvable network services, manageability and economic viability. The design is also informed by technology factors such as radio spectrum scarcity, wired bandwidth abundance, continuing Moore's law improvements to computing, and energy constraints in mobile and sensor devices.

The key components of the MobilityFirst network architecture are: (1) separation of naming and addressing, implemented via a fast global dynamic name resolution service; (2) self-certifying public key network addresses to support strong authentication and security; (3) generalized delay-tolerant routing with in-network storage for packets in transit; (4) flat-label internetwork routing with public key addresses; (5) hop-by-hop transport protocols operating over path segments rather than an end-end path; (6) a separate network management plane that provides enhanced visibility; (7) optional privacy features for user and location data; (8) content- and context-aware network services; and (8) an integrated computing and storage layer at routers to support programmability and evolution of enhanced network services. The architecture as a whole has been designed to be implementable with reasonable complexity, and to offer good scalability and performance. Although the proposed design has its "sweet spot" in large-scale mobile networking, its innovations and benefits will be enjoyed within the wired core as well, via enhanced security and robustness.

This project is a collaborative effort involving Rutgers, UMass, MIT, Duke, U Michigan, UNC, U Wisconsin, and U Nebraska with interaction with several industrial research partners. The project scope includes a progression of experimental prototypes: (i) individual validations of key protocol components such as name service, GDTN routing and flat-label interdomain routing; (ii) small-scale lab prototype of the architecture for controlled experiments; and (iii) Multi-site, medium-scale system prototype (using GENI infrastructure) for inter-networking experiments and proof-of-concept demonstrations. The project will conclude with a comprehensive validation and evaluation of the performance and usability of the architecture using both controlled experiments and application trials with real-world end-users. In the following sections we provide an early view of work-in-progress aimed at design and prototype validation of the MobilityFirst architecture.

Research supported by NSF FIA (Future Internet Architecture) grant #CNS- 1040735

II. ARCHITECTURE SUMMARY

The major design goals for the MobilityFirst architecture include the following: seamless user and device mobility; ad-hoc network formation and network mobility; tolerance to bandwidth variation and disconnection; multicast, multi-homing and multi-path support; context and content services; high throughput and spectral efficiency in wireless edge networks; strong security and privacy; usability and manageability. These requirements are achieved using the following protocol components:

1) *Clean separation between identity and network location*: MobilityFirst cleanly separates human-readable names, globally unique identifiers, and network location information [1-5]. The name certification service (NCS) securely binds a human-readable name to a globally unique identifier (GUID). A global name resolution service (GNRS) securely maps the GUID to a network address (NA). By allowing the GUID to be a cryptographically verifiable identifier (e.g., a public key or hash thereof), MobilityFirst improves trustworthiness; conversely, by cleanly separating network location information (NA) from the identity (GUID), MobilityFirst allows seamless mobility at scale.

2) *Decentralized name certification service (NCS)*: MobilityFirst decentralizes trust in name certification, i.e., different independent NCS organizations could attest to the binding between a human-readable name and the corresponding (public key) GUID. It is conceivable that the different organizations may disagree on the GUID corresponding to a name. End-users can choose which NCS(es) to trust and use quorum-based techniques to resolve disagreement between NCSs.

3) *Massively scalable global name resolution service (GNRS)*: The GNRS is one of the most central components of MobilityFirst and is responsible for supporting seamless mobility at scale. The scale we envision is on the order of 10 billion mobile devices moving through about 100 networks each day, which corresponds to an update overhead of ~10 million/sec. In comparison, DNS, by design, relies heavily on caching and takes on the order of several days to update a record. Thus, designing a massively scalable distributed GNRS is a key challenge in MobilityFirst.

4) *Generalized Storage Aware Routing*: MobilityFirst exploits in-network storage at routers in order to cope with variations in wireless access network bandwidth and occasional disconnections that inevitably occur in real-world mobility scenarios. Earlier work on the CNF architecture [6,7] demonstrated the benefit of storage and outlined storage-aware routing algorithms which consider long- and short-term path quality metrics while making forwarding decisions. The GSTAR protocol [8] further integrates delay tolerant networking (DTN) capabilities with CNF-like storage routing to provide a seamless solution for a wide range of wireless access scenarios.

5) *Content- and context-aware services:* The network layer in MobilityFirst is designed to be content-aware, i.e., it actively assists in content retrieval as opposed to simply providing a primitive to send packets to specified destinations in today's Internet. MobilityFirst achieves this by assigning GUIDs to content. These GUIDs are cryptographically verifiable, e.g., self-certifying hashes of the content, which allows a receiver to easily check the integrity of the content. MobilityFirst also extends the basic device and content GUIDs to more flexible groups of devices or users, e.g., all mobile devices in Central Park; or all taxis in Times Square, etc.

6) *Computing and storage layer*: Experience with the current Internet shows that it is imperative to design for evolvability. To this end, MobilityFirst routers explicitly support a computing and storage layer that enable rapid introduction of new, and possibly niche, services while minimally impacting the performance of the large majority of existing users.

III. PROTOCOL DESIGN

Based on the considerations outlined in Sec II we have developed an initial specification for the high-level protocol architecture of the network.

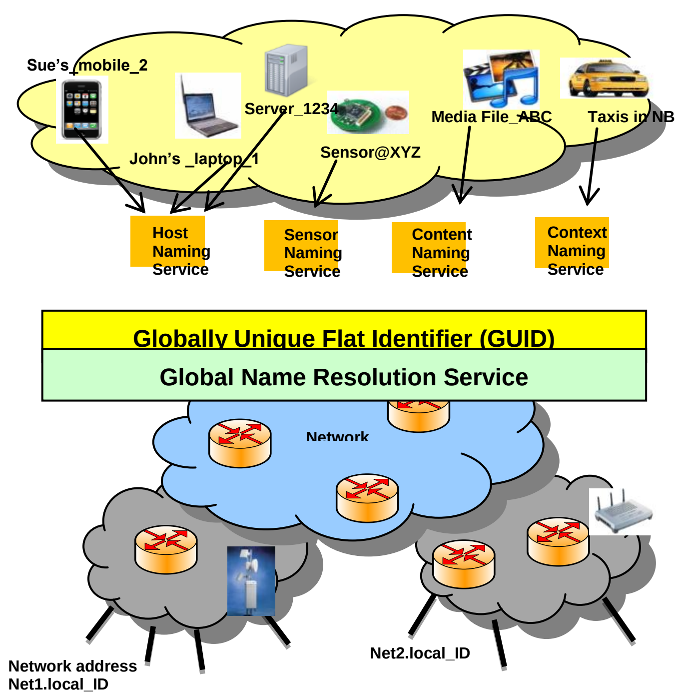

Fig. 1 Separation of Names from Addresses in the MobilityFirst Architecture

The MobilityFirst protocol architecture is based on a clean separation of names (of network-attached objects) from network addresses. As shown in Fig 1 above, there are a number of application specific "name certification" services which translate a human readable name such as "sensor@xyz" or "John's laptop" to a set of network addresses corresponding to the current points of attachment of the network object. The name service translates the human readable name to a unique GUID (globally unique identifier) which is used as the authoritative identifier for network attached objects, which may be devices, content, sensors, etc. The GUID is also a public key, thus providing a mechanism for authentication and management of trust for all network-attached devices or objects. This framework also supports the concept of context-based descriptors such as "*taxis in New Brunswick*" which can be resolved by a context naming service to a particular GUID which serves as a dynamic multicast group for all taxis currently in that area. Once a GUID has been assigned to a network object, there is an additional mapping from GUID to network address (NA) as shown in the figure. The idea is to assign routable network addresses to "network ports" and dynamically bind GUIDs to network ports using a new distributed network service called the "global name resolution service (GNRS)". The GNRS supports dynamic mobility simply by providing the current point of attachment of the mobile device, without the need for routing-level indirection associated with current networking protocols such as mobile IP. The network addresses (NAs) are expected to change at a slower time-scale and can use a second distributed network protocol (analogous to BGP in the Internet) for dissemination of routing updates.

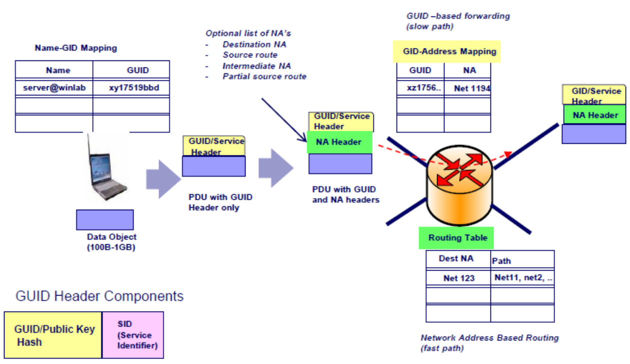

Fig. 2. Hybrid GUID/NA packet headers in MobilityFirst

Referring to Fig 2, it is observed that packets entering the network have the destination (and source) GUID attached to the protocol data unit (PDU). There is also a service identifier (SID) in the packet header which indicates the type of service required by the PDU including options such as unicast, multicast, anycast, context delivery, content query, etc. At the first access router, the destination GUID is resolved by accessing the global name resolution service (GNRS) which provides a dynamic mapping between GUIDs and routable NAs. The resolved NAs are optionally appended to the packet header thus making it possible for subsequent routers along the path to forward the PDU based on NAs alone – this is referred to as "fast path" forwarding. Any router along the path has the option of resolving the GUID again by querying the GNRS – this is the so-called "slow path" which allows for rebinding to a new set of NA's that may have resulted from mobility or temporary disconnection. The GUID routing option makes it possible to implement "late binding" algorithms where the decision on which network ports to route to is deferred until the PDU is close to the destination.

It is also noted that in the MobilityFirst architecture, PDU's may be large units ~100MB - ~1GB corresponding to complete audio or video files, and these are transferred as contiguous units from one router to the next. Another key feature of the architecture is the existence of in-network storage at routers. This enables us to use storage-aware routing protocols which have the option of temporarily storing PDUs at routers instead of forwarding towards the destination in order to deal with poor link quality or disconnection. A reliable hop-by-hop transport protocol is used to deliver packets between routers in contrast to the end-to-end approach used in TCP/IP.

Another key feature of the proposed MobilityFirst protocol stack is the service flexibility, with particular emphasis on multicasting, anycasting, multi-path and multi-homing modes as integral capabilities of the routing protocol. These service features have been provided in response to the needs of mobility applications which often care more about the context (e.g. device location or function) than its network address.

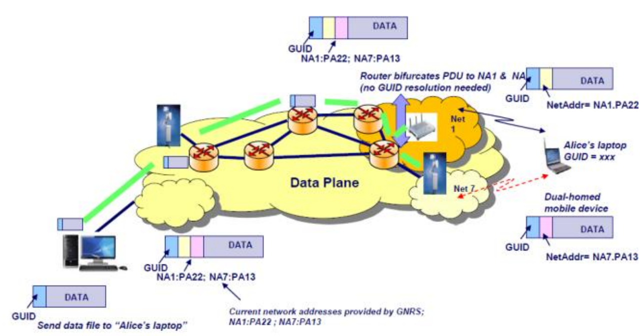

Fig. 3. Supporting Dual-Homing in MobilityFirst Routing

The GUID mechanism outlined above allows for context and content addressability with multicasting or anycasting to the set of network addresses associated with a GUID (such as *taxis in New Brunswick* or *Alice's_laptop*). A particularly interesting use case that is difficult to handle with conventional IP is that of "dual-homing" where a user's laptop may have two or more wireless interfaces (such as WiFi and 3G) on separate access networks, and the service objective is to deliver to at least one of these interfaces based on a suitable cost metric. An example of how the protocol works for such a dual-homing scenario is given in Fig. 3 above. In this example, the GUID for "Alice's_laptop" is resolved to two distinct network addresses corresponding to the 3G and WiFi networks that it is currently connected to. The PDU carries both these network addresses and the network routing protocol implements a "longest common path" type of algorithm before bifurcating the same message to both destinations.

IV. MobilityFirst Prototype on GENI

An early proof-of-concept prototype of the MobilityFirst architecture is currently under development, and was first shown at the GENI Engineering Conference-12, Kansas City in Nov 2011. This prototype is initially based on modified Click modular routers with additional GNRS and storage routing (GSTAR) functionality. The protocol stack to be implemented is shown in Fig 4 below.

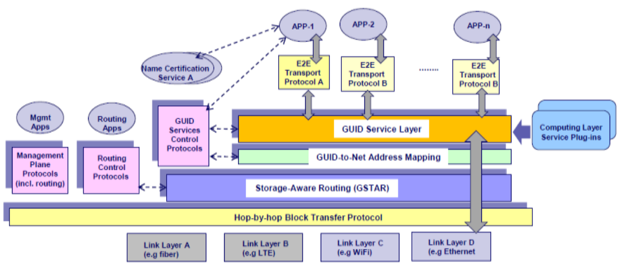

Fig 4. MobilityFirst Protocol Stack

This implementation of the protocol stack includes a hop-by-hop reliable link protocol based on the"HOP" protocol from UMass [9]. The GUID service layer has also been implemented on Linux and Android mobile clients, with a new set of service API's which correspond to the block delivery (both unicast and multicast), anycast and content query services offered by the new protocol. The Click modular router [10] implementation along with additional modules is outlined in Fig. 5 below. A two-level abstraction with fast-path as Click forwarding modules and slow-path as user level services (such as control and management) has been implemented.

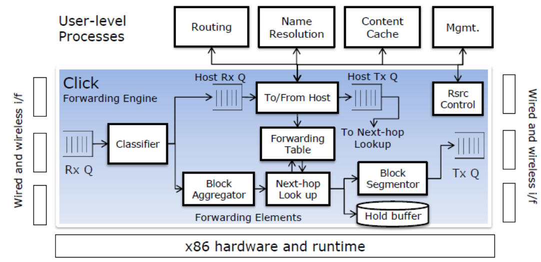

Fig 5. Click Modulator Router Implementation

The experimental system prototyped and demonstrated at GEC-12 in Kansas City consists of 7 programmable ProtoGENI routers spread across the US with two edge networks (located at BBN, Cambridge, MA and WINLAB, Rutgers, North Brunswick, NJ) with both WiMAX base stations and WiFi access points for end-user mobile access. The network's topology is illustrated in Fig. 6 below.

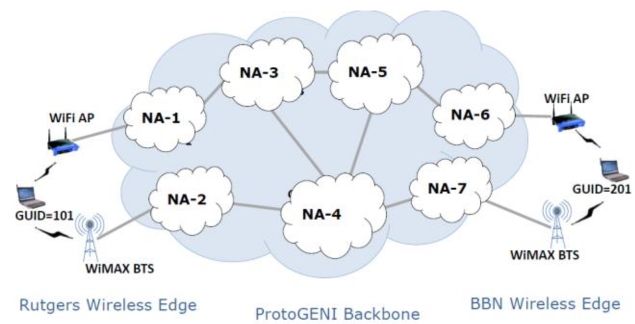

Fig 6. GENI Proof-of-Concept Experiment Configuration

The corresponding physical topology of the GENI nodes in the network uses a combination of different OpenFlow switches [11] and ProtoGENI nodes located at various sites across the US. The configuration used thus includes realistic RTT delays between router nodes, along with a variety of link speeds and access network techologies. In the scenario being considered, there are two dual-homed mobile devices in the BBN and Rutgers networks respectively, one serving as a content server and the other as a client. The GSTAR protocol provides features for efficiently delivering the content from server to client in a hop-by-hop manner with in-network storage and multi-homing (taking advantage of both the available WiMAX and WiFi interfaces).

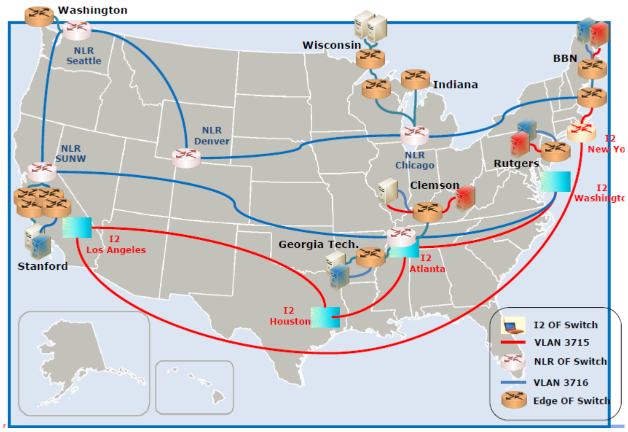

Fig 7. GENI Proof-of-Concept Experiment Configuration

The application demonstrated is a content delivery scenario (see Fig 8) in which content is labeled with a unique GUID, and then delivered in request to a "*get(GUID)*" query from the client. The content delivery operation involves GNRS resolution followed by GSTAR routing of the entire protocol data unit (media file) on a hop-by-hop basis. The experiment also demonstrates the fact that the GSTAR protocol is capable of dealing with edge network bandwidth variation and occasional disconnection that normally occur in the WiMAX and WiFi edge network. In particular, router nodes store the delivered blocks until channel conditions improve sufficiently for forwarding to resume along the path. The protocol is also capable of automatically choosing one or both available multi-homed paths in order to meet desired service and policy objectives.

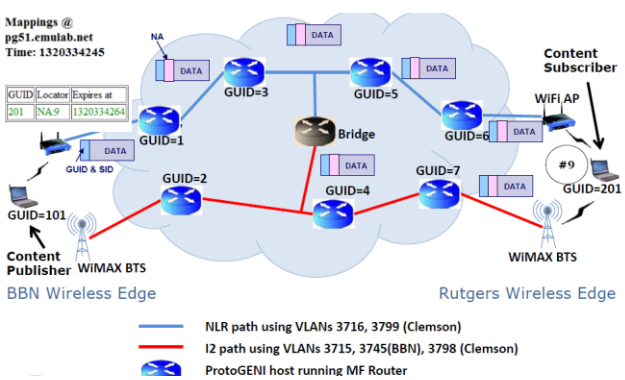

Fig 8. Content Delivery Application in GENI Experiment

V. Concluding Remarks

In this paper, we have presented an overview of the MobilityFirst future Internet architecture currently under development under the NSF FIA program. The design includes several key concepts for mobility-centric networks including separation of name and address, global name resolution service (GNRS), generalized storage-aware routing (GSTAR) and routing layer support for multi-path, multicast and multi-homing. A proof-of-concept prototype for the MobilityFirst protocol stack has been successfully developed on the GENI experimental testbed and demonstrated in Nov 2011. The results so far are promising, and we expect to report more definitive results with protocol details and performance results over the next 1-2 years. Further work will include inter-domain routing aspects, designs for content- and context-aware services, management plane features, and computing layer services.

VI. References

[1] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: a scalable peer-to-peer lookup protocol for Internet applications. IEEE/ACM Trans. Networks., 11:17–32, February 2003.
[2] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content-addressable network. In Proceedings of ACM SIGCOMM '01, pages 161–172, 2001.
[3] S. Deering, R. Hinden. Internet Protocol, Version 6 (IPv6) Specification. IETF RFC 1883, Dec. 1995.
[4] D. G. Andersen, H. Balakrishnan, N. Feamster, T. Koponen, D. Moon, and S. Shenker. Accountable Internet Protocol (AIP). In Proceedings of ACM SIGCOMM '08, Aug. 2008.
[5] R. Moskowitz and P. Nikander. Host Identity Protocol (HIP) Architecture. IETF RFC 4423, May 2006.
[6] Sanjoy Paul, Roy Yates, Dipankar Raychaudhuri, and Jim Kurose. "The Cache-And-Forward Network Architecture for Efficient Mobile Content Delivery Services in the Future Internet." Proc. of the First ITU-T Kaleidoscope Academic Conference on Innovations in NGN: Future Network and Services, 2008.
[7] Snehapreethi Gopinath, Shweta Jain, Shivesh Makharia and Dipankar Raychaudhuri."An Experimental Study of the Cache-and-Forward Network Architecture in Multi-hop Wireless Scenarios." Proc. Local and Metropolitan Area Networks (LANMAN), 2010.
[8] S. Nelson, G. Bhanage, and D. Raychaudhuri. "GSTAR: Generalized Storage-Aware Routing for MobilityFirst in the Future Internet Architecture." Proc. of ACM MobiArch Workshop, 2011.
[9] Ming Li, Devesh Agrawal, Deepak Ganesan, and Arun Venkataramani, "Block-Switching: A New Paradigm for Wireless Transport", USENIX NSDI 2009.
[10] Click Modular Router Project - http://read.cs.ucla.edu/click/
[11] OpenFlow Switching Specification. http://www.openflow.org/