A ID/Locator Separation Prototype Using Drone for Future Network

Shoushou Ren, Yongtao Zhang
2012, Network Technology Lab, Huawei Technologies Co., Ltd., Beijing, China
E-mail: {renshoushou, zhangyongtao3}@huawei.com

***Abstract*—The routing and addressing system of today's Internet is facing serious scaling problems, which are mainly caused by the overloading of IP address semantics. To address this problem, several recent schemes have been proposed to replace the IP namespace with separation of namespaces for identities and locators. ID Oriented Networks (ION) is one such mechanism. In this paper, a drone prototype based on ION implementation is described. An ID-to-ID communication between a moving drone and a stationary endpoint is demonstrated. ION protocol primitives are defined along with packet format, encapsulation/decapsulation, as well as the handover process. The results obtained from the prototype of ION show that the ID-to-ID communication continues to works well and is not interrupted when the location of the drone changes. This prototype shows that the basic idea of ID/Locator separation is a feasible and positive way to solve the scaling issue in the current Internet Protocol.**

Keywords-drone; identifier; locator; handover.

I. INTRODUCTION

It has been widely recognized that today's Internet routing and addressing system is facing serious scaling problems [1][2]. A common consensus is that this scaling issue is mainly caused by the overloading of Internet protocol (IP) address semantics [3]. That is, an IP address represents not only the location but also the identity of a host. Therefore, several new schemes [4], such as the Locator/ID Separation Protocol (LISP) [5] and Host Identity Protocol (HIP) [6][7][8], have been proposed to replace the IP namespace in today's Internet with a locator namespace and an identity namespace. In these schemes, a locator namespace consists of *locators* that represent the attachment point of hosts in the network, while the identity namespace consists of *identifiers* (ID), also known as endpoint identities (EIDs) that represent unique identities of hosts. When IDs are separated from their network attachment position information, packets destined for IDs are generally forwarded with the default routing method by using the locators as IPs. By decoupling an identifier from its locator, changes to a host's location become transparent to the upper layers above including TCP.

Consider the communication between two User Equipments (UEs) in the ION network. Each UE only needs to know the other's ID before the connection is established, since only the ID can tell them *who* the correspondent ID is. While the locator is only used for packet forwarding in the internet and it may change according to different access gateways. Thus, the communication is called an ID-to-ID communication.

In this paper, we present a drone prototype which is realized based on the basic idea of ION. The drone has a unique and fixed ID when flying across different access gateways. While its locator changes when it flies across the network accessing different gateways. Our prototype ensures that the drone can establish an ID-to-ID connection with the remote ground station, which is also an ID aware host. Moreover, when the drone accesses different gateways, the ID-to-ID communication between the drone and the ground station is continuously maintained even when the drone's locator changes.

The rest of this paper is structured as follows. In Section II, we introduce the basic framework of the Identity Oriented Network. In Section III, we describe the topology of the drone prototype and introduce the main entities in the prototype. In Section IV, the detail designs of our prototype are presented, including the id packet format, packet encapsulation and decapsulation, as well as the handover process. At last, we conclude this paper in Section V.

II. IDENTITY ORIENTED NETWORKS

Based on the idea of Identity and Location separation, ION framework is briefly described in Figure 1 and the details are out of scope for this paper. Since identity and locators are separated, ION expands network layer concept to accommodate ID in the following manner.

- *ID layer* is a distributed function responsible for ID management and authentication services.
- *Mapping system:* An ID/location resolution system is introduced which maintains mappings between a host and its location.
- *ID based connection:* In order to inter-connect two endpoints independent of network address an ID aware socket connection.

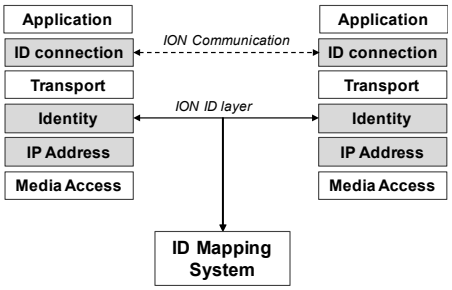

Figure 1. Brief framework of ION.

 ISBN: 978-1-61208-550-0

ION architecture enhances traditional network layer with identity awareness. Some advantages ION scheme include (a) communication of non-IP devices such as IoT, (b) a smoother and seamless location agnostic mobility and (c) cross-silo communication across applications working with same network entities. Please refer to Next Generation Protocols (NGP) paper for further details on ION [9].

III. TOPOLOGY OF THE DRONE PROTOTYPE

The topology of our drone prototype is depicted in Figure 2, which mainly consists of following five entities:

- *Universal Access Gateway (UAG):* The UAG is the edge access gateway in the ION architecture. The UAG is in charge of locator assignment, locator management and access control. When a UE, such as the drone in our prototype, is online and accesses to a UAG first time, the UAG assigns an IPv6 address as locator to it. Then the UAG registers the ID/Locator mapping item of the UE to the mapping system and caches the item until the UE leaves. The UAG can support the wired access as well as wireless access of UEs. UAGs also perform packet forwarding function as traditional gateways. Three UAGs are deployed in our prototype and the drone flies randomly in the area covered by the three UAGs.
- *Access Point (AP):* Traditional APs. The drone access to the UAG via an AP. Only one AP is deployed under each UAG for the case of layer-3 handover [10] [11], which will be further explained in the next section.
- *Drone:* The drone is an ID aware host with a unique and fixed ID. When it accesses a UAG, a locator will be assigned, which is used to locate where it is. The drone is equipped with a camera for shooting real-time video when flying across different UAGs. It is controlled by the ground station and the video will be transmitted to the ground station via ID-to-ID communication.
- *Ground Station (GS):* the GS, which is also an ID aware host, is the controller of the drone. It receives and displays the video shot by the drone.
- *ID-Locator Mapping System (ILMS):* The ILMS stores all the ID/Locator mapping items that have been registered. Once a UE is assigned a locater or by its access UAG, the ID/Locator item will be registered or updated to the ILMS. If a UE wants to communicate with other ID hosts, their locators can also be retrieved from the ILMS.

Note that ID of hosts may be set before leaving the factory or assigned after that by some organizations. In our prototype, we use the IPv6 address those are with prefix *2F00::* as IDs. The goal of our prototype is: 1) realize an ID-to-ID communication between the drone and the remote GS; 2) when the drone's locator changes while roaming across different UAGs, the ID-to-ID communication could be kept continuous

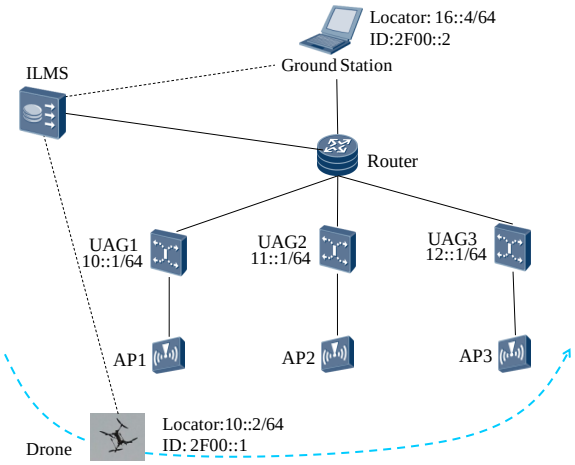

Figure 2. Topology of the drone prototype.

IV. PROTOCOL PRINCIPLE

Some new protocol principles are designed to realize the ID-to-ID communication between the drone and GS.

A. Packet Format

The main change in ID packet lies in the IP-layer header. The tuple <*src_ip, dst_ip*> in a normal IP packet is replaced by a new header of tuple <*src_id, dst_id, src_loc, dst_loc*> in the id packet, which is shown in Figure 3. In this prototype, the IP address in the normal IP packets has the same meaning with the locator in id packets.

B. Packet Encapsulation

The packet encapsulation process of id packet in the id-to-id communication is depicted in Figure 4.

When a packet is generated by the TCP layer, it will be first checked by an *is_ID()* function to determine whether it belongs to an ID-to-ID communication based on its *src_ip* and *dst_ip*, which can be found in the 5-tuple of TCP sockets. If the *src_ip* or *dst_ip* is with IPv6 prefix *2F00*, the packet will be further encapsulated into an id packet by the *id_out()* function. Otherwise, the packet will be sent to the *dst_ip* as a normal IP packet.

If the *2F00* prefix is detected, the drone tries to get the locator of the GS in its own cache and the UAG's cache. If fails, a request will be sent to the ILMS for the retrieval of GS's locator according to its id. Then, the normal packet will be encapsulated as Figure 3 shows. The drone's locator, i.e., the *src_loc*, is assigned when it accesses a UAG. The *dst_loc* is retrieved from caches or from the ILMS. Since we use the ipv6 address with prefix *2F00* as id, the *src_id* in id packet is the same with *src_ip* in the normal packet, and the *dst_id* in id packet is the same with *dst_ip* in the normal packet.

 ISBN: 978-1-61208-550-0

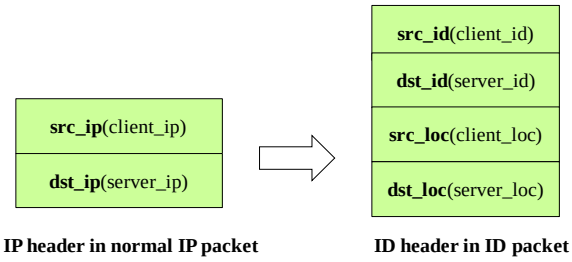

Figure 3. Changes of IP header in ID packet.

At last, the encapsulated ID packet will be sent to the access AP and UAG. The access UAG just treats the locator as the normal IP and forwards all packets as usual according to its routing table.

C. Packet Decapsulation

The decapsulation process of ID packets is shown in Figure 5. Once a packet is received by the hardware, it will be send to the IP layer and checked by the *is_ID()* function to determine whether it's an id packet or not. If the packet is a normal packet, it will be sent to the TCP layer directly. Otherwise, it will be treated as an id packet and further decapsulated by the *id_in()* function. The *id_in()* strips the locator header, *src_loc* and *dst_loc* fields. Then the stripped packet will be sent to the TCP layer as a normal packet.

It should be noted that in this prototype, the ID hosts (i.e., the drone and the GS) are designed to be aware of ID/locator separation. The locator header of ID packets is encapsulated and decapsulated at the drone for realization convenience. In fact, the ID/locator separation network can also be designed as that the hosts are completely unaware of ID/locator separation. This can be realized by embedding the encapsulation and decapsualation of id packets into gateways rather than hosts.

D. Handover

When the drone moves outside the range of its access AP, a handover process must be handled. Since the layer-2 handover [12][13] doesn't lead to changes of locator, we only consider the layer-3 handover in this prototype. Only one AP is deployed under each UAG, which means when the drone flies across different APs, its locator will change, leading to a layer-3 handover.

The layer-3 handover process is detailed in Figure 6.

Step 0: the drone, with id *2F00::1* and locator *10::2* assigned by UAG, communicate with the GS, whose id is *2F00::2*, via UAG1.

Step 1: The drone probes the signal strength of the access AP. Once it detects the signal strength is lower than a threshold, the handover process will be activated. Then the drone sends a handover notification to UAG1.

Step 2: Upon receiving the notification, UAG1 will send a confirm information to the drone and starts to caches packets with *dst_loc* or *des_ip* equals to *10::2*.

Step 3: After receiving the confirmation from UAG1, the drone disconnects from the UAG1-AP and tries to connect the AP under UAG2. If success, the drone will get a new locator *11::2*, which is assigned by UAG2. Then the drone uses the new locator to notify the ILMS as well as the GS that its locator has changed from *10::2* to *11::2*. The ILMS and the GS then update their mapping item related to id *2F00::1* and return the confirmation to the drone that its locator has been updated. At the same time with sending the locator update notification, the GS will also send its new locator to UAG1, notifying UAG1 that it has successfully finished the handover and requests for the cached packets. Upon receiving the notification, UAG1 also sends a confirmation to the drone.

Step4: With the same id *2F00::1* and the new locator *11::2*, the drone continues the id-to-id communication with the GS. The packets in fly will also be tunneled to the drone according to the new locator.

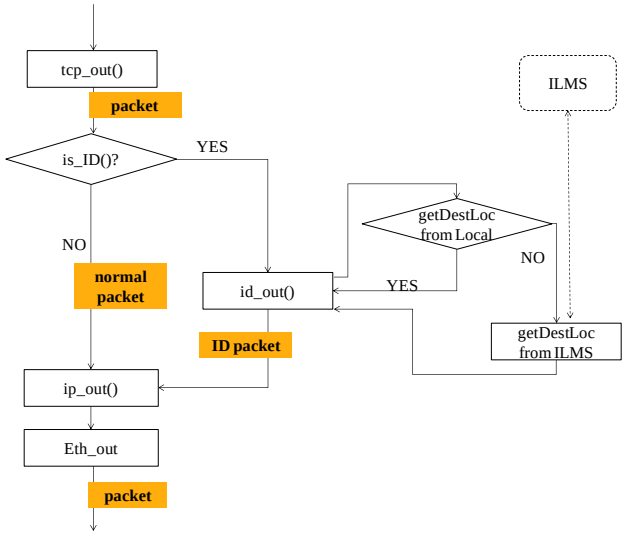

Figure 4. Packet encapsulation process.

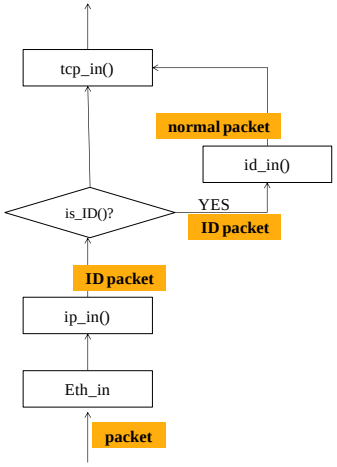

Figure 5. Packet decapsulation process.

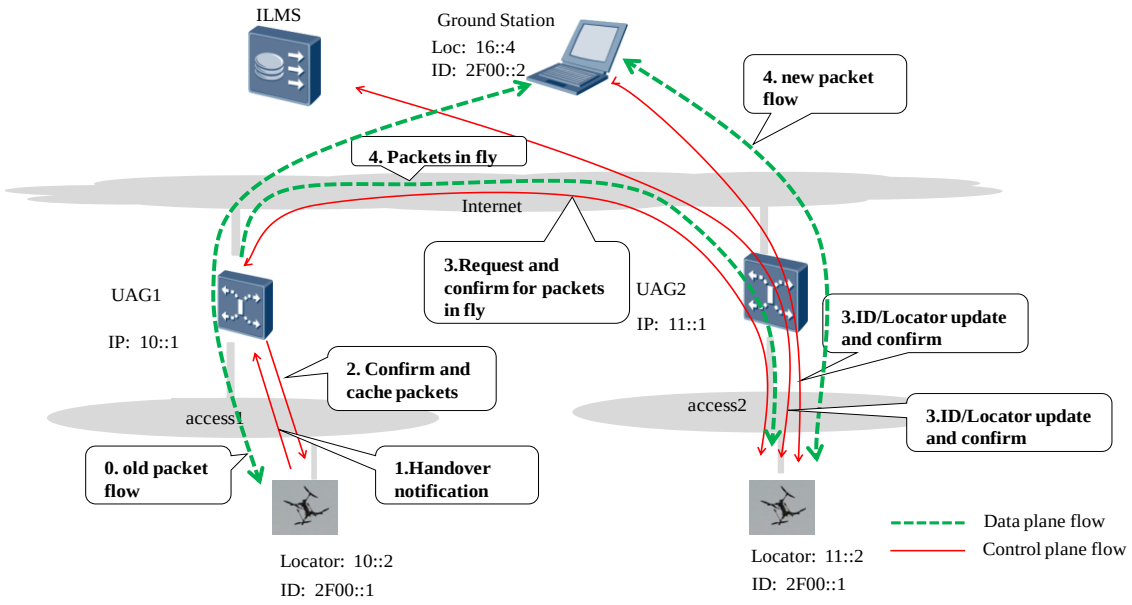

Figure 6. Handover process in id-to-id communication.

From the view of the GS, the corresponding node in the ID-to-ID communication is always the drone during the handover process. Thus, changes of the drone's location is transparent to the upper layers above including TCP/IP, and the ID-to-ID connection can be kept continuous.

V. CONCLUSION

In this paper, we presented a drone prototype based on the idea of ID/Locator separation in the ION. ID is designed as the only identifier of hosts, while the locator is only used for routing and packet forwarding. ID-to-ID communication is realized between the drone and the ground station. We also proposed some protocol principles to define the format, as well as encapsulation/decapsulation of id packets. The handover process is also designed.

The basic idea of ID/Locator separation is now widely accepted by researchers and Internet organizations such as IETF. This prototype shows that this basic idea is a feasible and positive way to solve the scaling issue in the current Internet Protocol.

REFERENCES

[1] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, "Overview and Principles of Internet Traffic Engineering". IETF Internet Standard, RFC 3272, May 2002.

[2] BGP Routing Table Analysis Reports, http://bgp.potaroo.iiet/, 2012.

[3] D. Meyer, L. Zhang, and K. Fall, "Report from the IAB Workshop on Routing and addressing". IETF Internet Standard, RFC4984, September 2007.

[4] R. Koodli, Ed., "Fast Handovers for Mobile IPv6", IETF Internet Standard, RFC4086, July 2005.

[5] D. Farinacci, V. Fuller, D. Meyer, and D.Lewis, "The Locator/ID Separation Protocol (LISP)," IETF Internet Standard, RFC6830, January 2013.

[6] R. Moskowitz and P. Nikander, "Host Identity Protocol (HIP) Architecture", IETF Internet Standard, RFC 4423, May 2006.

[7] R. Moskowitz, P. Nikander, P. Jokela, and T. Henderson, "Host Identity Protocol, IETF Internet Standard", RFC5201, April 2008.

[8] Henderson, T. R., Ahrenholz, J. M., and Kim, J. H., "Experience with the host identity protocol for secure host mobility and multihoming," In IEEE Wireless Communications and Networking, pp. 2120-2125 ,2003.

[9] DGS/NGP-004, "Next Generation Protocols: Evolved Architecture for mobility using Identity Oriented Networks".

[10] D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," IETF RFC 3775, June 2004.

[11] R. Koodli, "Fast Handovers for Mobile IPv6," IETF RFC 4068, July 2005.

[12] H. Soliman, C. Castelluccia, K. El Marlki, and L. Bellier, "Hierarchical Mobile IPv6 Mobility management," IETF RFC 5380, Oct. 2008.

[13] H. Y. Jung, H. Soliman, S. J. Koh, and J. Y. Lee, "Fast Handover for Hierarchical MIPv6," IETF Internet Draft, April 2005.

 ISBN: 978-1-61208-550-0