Introduction

Software Defined Networking (SDN) brings a new perspective to computer networking where control plane is sketched as a logically centralized entity isolated from the data plane. This logically centralized control simplifies provisioning and management of network services while enables rapid integration of new protocols.

[Difference between SDN and legacy IP network]
In traditional networks, control plane is co-located with the forwarding hardware. In other words, control plane is distributed in legacy IP networks. This distributed architecture requires provisioning every network device by the network administrator to deploy a certain network service. On the other hand, in SDN, control functionality is logically centralized and new services can be deployed by provisioning an application instead of having to configure every data plane switch. Moreover, configuring network devices in traditional networks require use of vendor specific commands. A script written to deploy a particular service on a certain vendor equipment is unlikely to be useful on a network with devices from another vendor. Software Defined Networks doesn't suffer from this problem because control plane is independent from forwarding infrastructure. Data plane device from any vendor must conform the the open southbound protocol standard and can be controlled by the SDN controller. Additionally, it is difficult to deploy a new protocol in IP legacy network. This is because in legacy network equipments are not designed to allow programmability of the control logic and even if these traditional devices could be programmed, every device in a network, potentially from different vendors, must be upgraded for a new protocol to work. In SDN, on the contrary, new protocols can be easily deployed by writing a controller application because every device in the network will automatically conform to the policies set by that application.

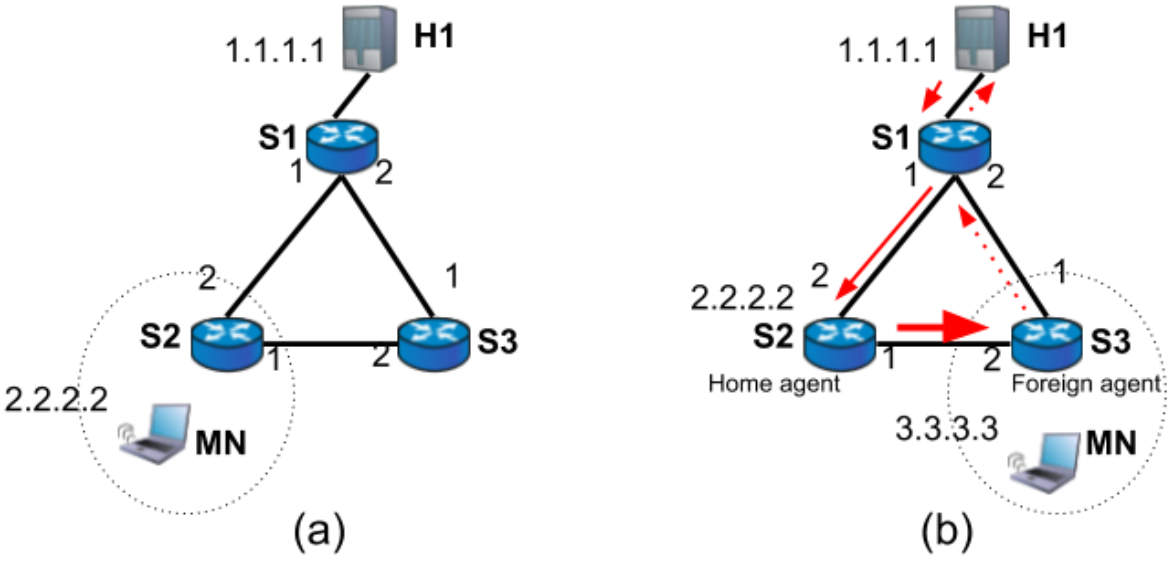

Fig. 1: Traffic forwarding in Mobile IP

[Advantages of deploying SDN in Mobile IP Network]

Mobile IP is used in IP networks with mobile nodes to deliver packets destined to a permanent IP address of the mobile node while allowing it to change it's point of attachment in the Internet and a have different IP temporary address than it's permanent IP address. This temporary address is called care-of-address. The permanent IP address belong the the home network of the mobile node. The mobile IP protocol takes care of informing the home network of the present care-of-address of the mobile node. An entity residing in the home network, called home agent, forwards the packets address to the permanent address of the mobile node to it's care-of-address. This way the remainder of the Internet doesn't have to be aware of the mobility of the node. As shown in Fig. 1, traffic from any host (called correspondent node) to the mobile node is first sent to the home agent. A tunnel (shown in thick red arrow) is created between the home agent and foreign agent. The home agent forwards traffic using this tunnel to the mobile node. The mobile node can directly send the return traffic to the correspondent node without using the home agent. This is called triangle routing. The mobile IP protocol introduces additional delay and overhead in the data communication due to the following reasons-

1) Traffic towards the mobile node takes a detour via the home agent which causes creates a longer path than it would be needed if packets could travel directly. As shown in Fig. 1(b), the traffic from H1 to MN travels through the path H1-S1-S2-S3-MN which is longer than the direct path H1-S1-S3-MN. This causes both additional delay and requires overall more bandwidth from the network by occupying more links than it would be needed with direct communication. In Fig. 1(b), the two links S1-S2 and S2-S3 are occupied for the traffic whereas it would be enough to only occupy only one link S1-S3. Moreover, the S2 might become a routing bottleneck as it is the common node for forwarding traffic for all mobile nodes for which it is the home agent.
2) The mobile node must obtain care-of-address from the foreign agent by listening to agent advertisement or using agent solicitation message or by means outside mobile IP (collocated care-of-address) which introduces delay.
3) The mobile node has to perform registration of its care-of-address with the home agent which introduces delay.
4) When a mobile node returns to its home network, it has to perform deregistration to prevent packets from being forwarded to the previous care-of-address which introduces delay.
5) Tunnelling used between home agent and foreign agent requires an extra header which increases bandwidth consumption.

In addition to the delay and bandwidth overhead introduced due to the above reasons, mobile IP fails to operate for clients that doesn't support gratuitous ARP. When a mobile node moves away from the home network and registers it's care-of-address with the home agent, home agent will create gratuitous ARP in the home network. But, hosts that doesn't support gratuitous ARP will continue to send traffic to the MAC address of the mobile node instead of towards the home agent.

The centralized control and programmability of SDN can be used to overcome the limitations of mobile IP. In this paper, we propose a protocol that avoid the delay and bandwidth

overhead caused by mobile IP. Our protocol allows direct communication between the mobile node (MN) and rest of Internet regardless of the point of attachment of the MN. In the proposed SDN based protocol, hosts in the home network can continue to communicate with the mobile node after the mobile node has moved away even if these hosts doesn't support gratuitous ARP. This is possible because SDN controller can set up necessary flow entries to forward traffic between the hosts in home network and the MN. We describe our protocol in the next section.

Protocol Description

In our protocol the mobile node (MN) owns it's IP address (IP) regardless it's point of attachment on the network. Because, the source IP address of the MN is owned by the MN, we can uniquely identify each MN by its IP address. This means, we do not require the concept of home address and care-of-address. The SDN controller will store a mapping of present location of each node in the form (IP, SW) where SW is the SDN switch of present attachment for the MN. The network will always route the datagrams destined to the MN at its present location. To do so, firstly, the SDN controller needs to be notified of the mobility events. Then, the flow rules need to be updated according to the new topology (where the MN has moved from the previous location to the new location).

To notify the SDN controller about the mobility of nodes, we use Packet-In message. When a SDN switch receives a new flow from a MN, it will generate a Packet-In message towards the SDN controller. With this Packet-In message, the SDN controller knows that the MN has moved away from its previous location. Our protocol requires that, source IP address is always used as a match field to create any flow entry.

Once, the SDN controller is notified about the location change of the MN, it will update its mapping (IP, SW) for that MN. The SDN controller also maintains list of switches for each MN. In this list the SDN controller keep track of the flow entries installed in different switches for each MN. On location update, these old flow entries will be updated when required to reflect the new location of the MN.

We distinguish the ports in an SDN switch as UNI (user network interface) and NNI (network-to-network interface). When installing flow rules to a switch, ingress interface will always be used as a match field. Using ingress interface and source IP address as match field for flow entries ensures that a Packet-In message will be generated upon location update even if the new switch previously had flow entry for the MN.

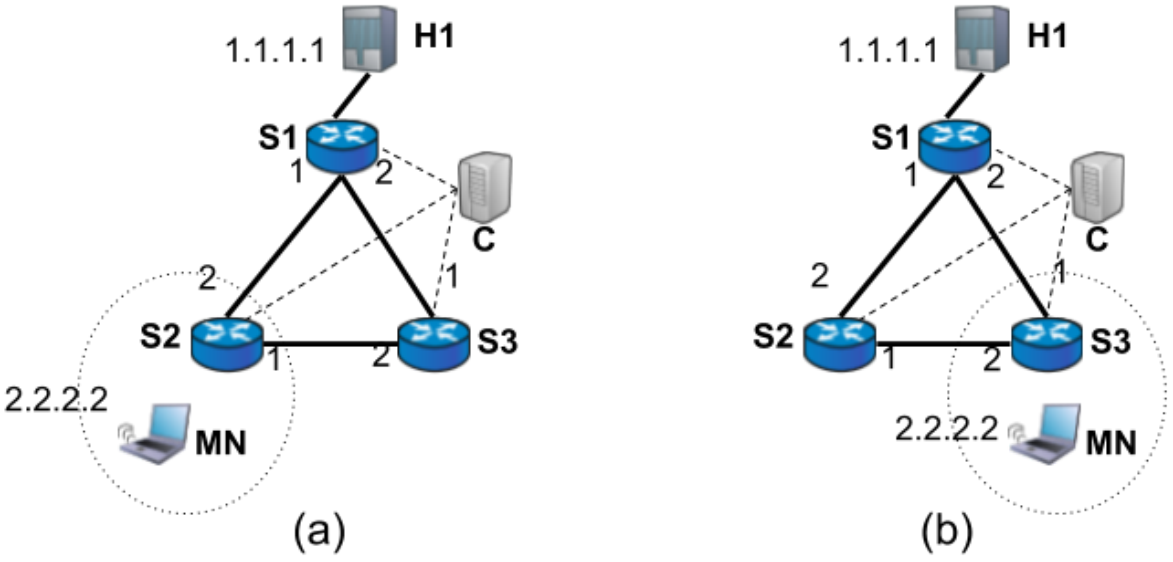

Fig. 2: Operation of the proposed protocol

In Fig. 1 (a), mobile node (MN) with IP address 2.2.2.2 is attached to switch S2 at its home network. The SDN controllers maps MN to S2 in its location map as (2.2.2.2, S2). The traffic from MN can reach the host H1 either using the path MN-S2-S1-H1 or the path MN-S2-S3-S1-H1. Let us first assume that the former path is in use. Now, when the MN moves to S3 in Fig. 1 (b), S3 will receive packets from MN. Because, S3 previously didn't have any flow entry with MN's IP address as the source IP, a Packet-In message will be generated. Now, let us assume that the later path (MN-S2-S3-S1-H1) was being used in Fig. 1 (a). In that case, S3 had a flow entry with MN's IP address as the source IP. But, when MN moves to S3 in Fig. 1 (b), there is no match with the flow entry for the path MN-S2-S3-S1-H1. This is because the existing path will have an ingress port of 2 where as the packet from MN has a different ingress port. This will ensure that a Packet-In message will still be created.

As soon as the Packet-In message is received, the SDN controller C will look at the list of switches where flow entries have been installed for MN. For example, for the path MN-S2-S1-H1, the switches are S2 and S1. To set up the new path, the SDN controller will delete flow entry for S2, update flow entry for S1 and create a new flow entry for S3. Updating flow entry at S1 is needed because the ingress port is different although the egress port, source IP and destination IP are same.

It is evident that, the traffic from MN can use any path to reach H1 as decided by the SDN controller (e.g. MN-S3-S1-H1) and doesn't have to travel its home network switch S2 with the proposed protocol.

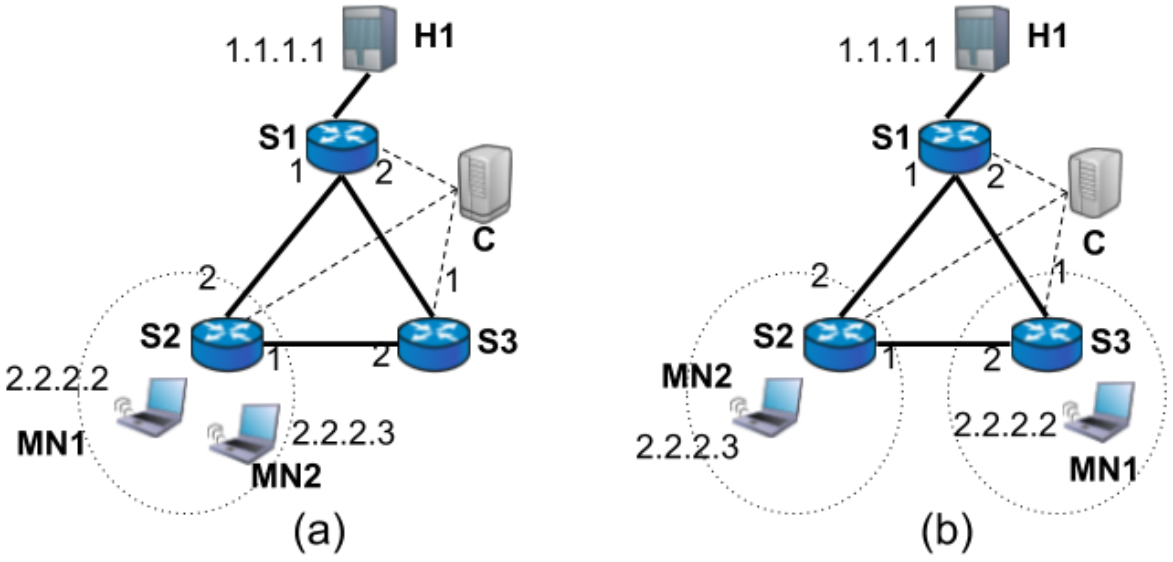

Fig. 3: Resolving ARP limitation of Mobile IP using SDN

Now, let us assume that there are two mobile nodes, MN1 and MN2, initially under S2 (Fig. 2a). MN1 and MN2 uses ARP to decide each others MAC address because their IP addresses fall under the same network. Now, if MN1 moves to S3 (Fig. 2b), MN2 will still send packets to MN1's MAC address instead of to S2's gateway address. When the Packet-In message is received from S3, the SDN controller C will replace the L2 flow entries (the flow entries for routing within the local network using only MAC addresses) at S2 with L3 flow entries that uses ingress port, source IP, destination IP etc. Therefore, our protocol will be able to seamlessly handle the traffic from MN2 even if MN2 doesn't support gratuitous ARP. In case of mobile IP, this wouldn't be possible.

We illustrate our protocol as a chronological list of events and show the communication between nodes in the network in Fig. 3.

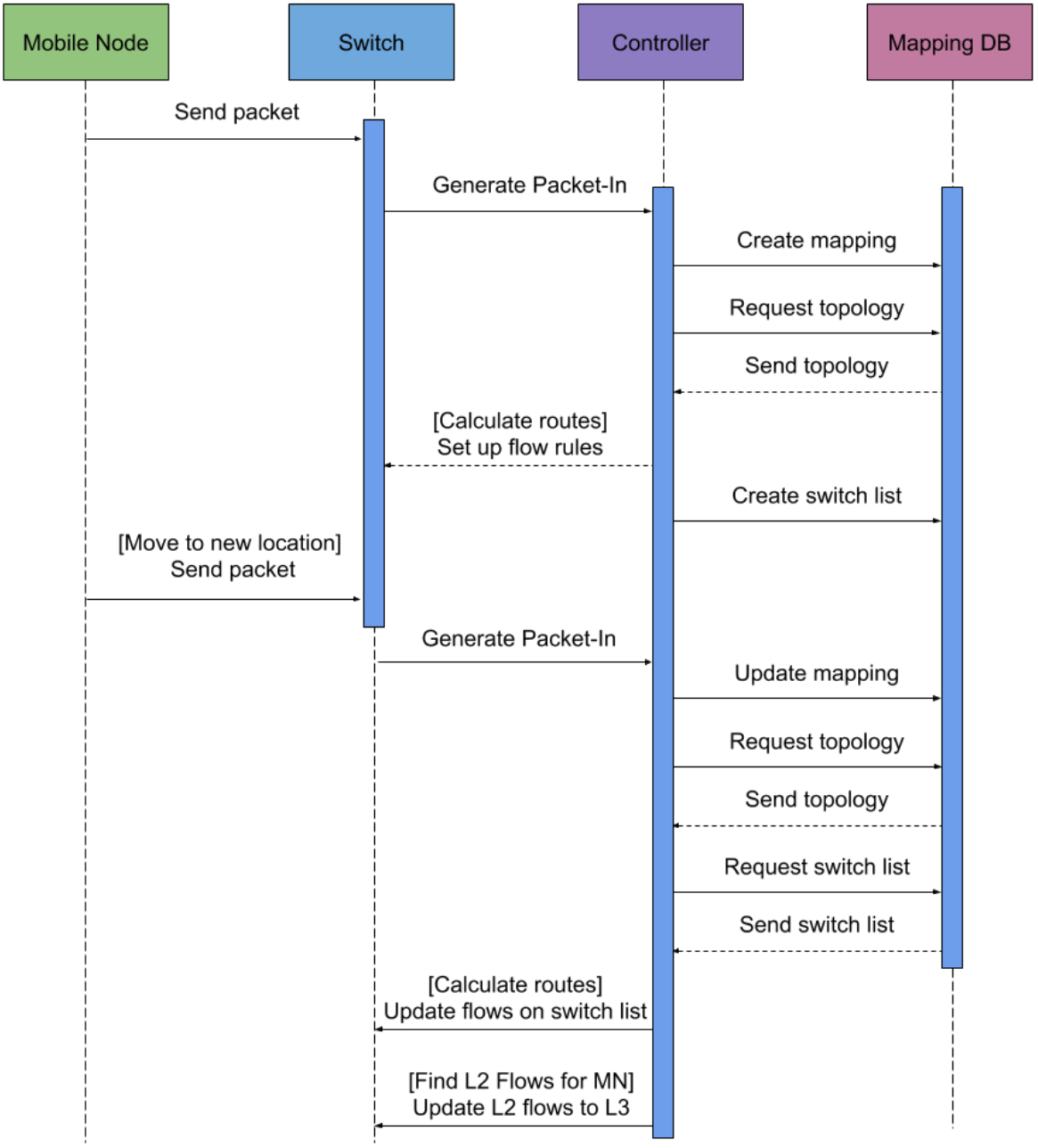

Fig. 4: Proposed protocol