

Deploying Virtual MAC Protocols Over a Shared Access Infrastructure Using MAClets

G. Bianchi¹, P. Gallo², D. Garlisi², F. Giuliano², F. Gringoli³, I. Tinnirello²

¹ CNIT / Università degli Studi di Roma - Tor Vergata, Italy

² CNIT / Università degli Studi di Palermo, Italy

³ CNIT / Università degli Studi di Brescia, Italy

Abstract—Network virtualization has been extensively researched in the last years as a key enabler for improving the network performance. However, virtualization in wireless networks pose some unique challenges: first, the usual over-provisioning approach for providing isolation between multiple virtual entities is not viable; second, the partitioning criteria are often ambiguous, since the actual resources perceived by each entity depend on many external (and time-varying) factors.

In this demo, we show an effective virtualization solution for wireless local area networks, solving the problem of isolation and flexible resource partitioning, based on the concept of MAClets. MAClets are software programs uploaded and executed on-demand over wireless cards, and devised to change the card’s real-time medium access operation. MAClets can be directly conveyed within data packets and executed on hard-coded devices acting as virtual MAC machines. A multi-operator virtualization experiment involving the distribution of MAClets within data packets, and their execution over commodity WLAN cards, shows the flexibility and viability of the proposed concept.

I. INTRODUCTION

The benefits of virtualization have been usually considered from the operator point of view and especially for cellular networks. However, several potential benefits also exist for local area networks and clients. First, running different logical adaptors (clients and access points) on a single hardware may avoid hardware underutilization and reduce costs and energy requirements. Second, especially in an unlicensed spectrum, virtualization can be used for assuring a better control of the wireless resources, isolating untrusted applications, or giving priority to different groups of users. Finally, virtualization can be a key enabler for implementing customized networking solutions, effective in specific niche contexts or particular network situations, as well as for fault and error containment [1].

A common practice of access point vendors for supporting virtualization is creating multiple logical networks (called multi-SSID) over the same radio on a single radio channel. Logical layer virtualization is achieved thanks to the broadcast of multiple beacons, announcing each of them a logical network with particular security and characteristics. At the client side, the MultiNet [2] architecture allows to create different logical clients on the same hardware interface, but with a significant switching time between networks (higher than 150ms in their implementation, due to the card hardware latencies). Another extension, able to support multiple logical planes on top of a single radio, has been represented by the FreeBSD project [3]. This implementation enabled the creating of multiple BSS planes with a single radio, including multiple logical Access Points, ad-hoc networks, or a combination of them. The architecture has also been extended in [4], that is the first successful implementation of simultaneous connections to multiple APs. The key idea is exploiting opportunistically power save mode for maintaining simultaneous associations with multiple APs.

While these proposals have been the starting point for a true virtualization, they are often implemented at the driver level (in the

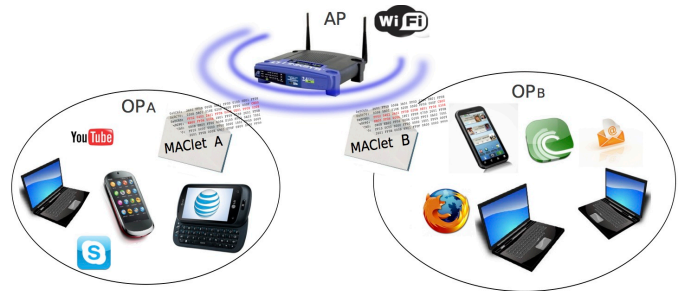


Fig. 1. Multi-operator virtualization scenario: the AP sends different MAClets (MAC state machines) to the stations associated to each operator.

middle between IP and MAC) and do not provide full virtualization, i.e. the possibility to employ different MAC schemes or parameters on different virtual interfaces. However, these constraints are not actually hardware constraints, but depend on the current card architecture which implement the so called low-level MAC operations (i.e. the time stringent medium access rules) inside the card. In this demo, we prove how to overcome these limits on an ultra-cheap commodity card (namely, the AirForce54G card by Broadcom), by replacing the card original firmware with a new one implementing a *MAC Engine* [5], i.e. an executor of generic state machines. The new card interface is then exploited for dynamically loading different MAC programs and freezing states, which dramatically simplify the management of programmable virtualization solutions.

II. FULL-MAC VIRTUALIZATION

Assume that the same Access Point (belonging to a public network) is shared between two different WiFi operators. The two operators want to implement a different service model: the first operator (operator A) advertises “FIXED” SSID, offering access to the Internet with a fixed (guaranteed) bandwidth, while the second one (operator B) advertises “BEST” SSID, offering a traditional best effort access.

Although the standard includes PCF and HCCA for managing the medium access by means of polling, the lack of support in commercial products prevents an easy solution which guarantees the coexistence of the two service models. Indeed, there is not way to isolate the two groups of stations associated to the different operators, in order to avoid that a transmission performed by a station in the first group does not take resources allocated for the second operator.

A. MAClets

Our virtualisation solution is based on the concept of MAClet [5], a *coded state machine* that together with an initial state and an activation event represents the high level description of MAC programs like TDMA, CSMA or an hybrid scheme. The MAClet is executed by the *Wireless MAC Processor* (WMP), a MAC-agnostic wireless engine that exposes a common Application Program Interface for sensing *events* and *conditions* and running desired elementary *actions*. In our

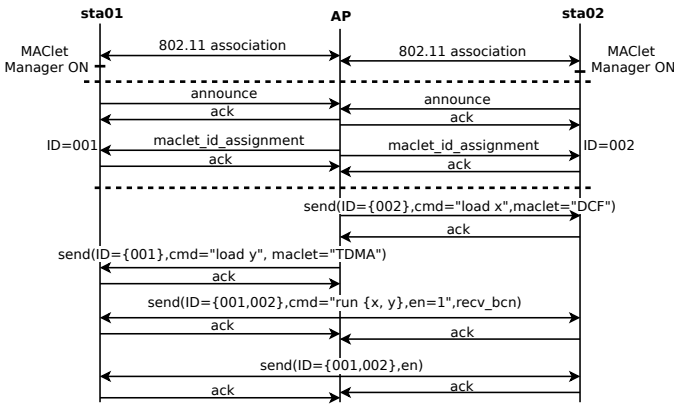


Fig. 2. Messages of the MAClet Distribution Protocol: an example.

vision, APs customize stations' behaviours by sending configuration packets that embed MAClets and specify which event should trigger the transition from one MAC program to another.

When stations are powered up, they load the MAClet that deploys the legacy DCF as default state machine: this allows stations to associate to an AP for receiving, when needed, the correct MAClet for the BSS they belong. MAClet activation can eventually be set asynchronously, if no event is specified, or upon the occurrence of the triggering signal like the reception of a control frame or the expiration of a relative or absolute timer: examples of MAClet Distribution and Synchronization mechanisms are discussed in [6]. We show in Figure 2 an example of messages exchanged between the AP and two stations for loading two different MAClets, a TDMA protocol on station 1 and a legacy DCF on station 2: here the activation of the two MAClets is triggered by the reception of the next beacon.

B. Demo Description

Resource repartition between two operators can be addressed in a very effective and flexible manner with MAClets. According to the SSID specified in the association request, each station receives a different MAClet: a DCF program for the stations associated to the "BEST" SSID, and a TDMA program for the stations associated to the "FIXED" SSID. The DCF MAClet is a legacy DCF program modified for being suspended at the reception of a new beacon, and reactivated by the expiration of a time-out that is set before suspension. The TDMA MAClet, on the contrary, is suspended first, and activated *after* the time-out. This mechanism guarantees a perfect coexistence and isolation between the two networks: stations connected to the same operator access the channel during the same time interval without interfering with stations belonging to the other operator. Moreover, the time reserved to each operator can be dynamically tuned (by updating the MAClet configuration parameters) according either to traffic conditions or to service level agreements between the two operators. Isolation is not obviously guaranteed with other external interfering networks.

The virtualization experiments can be tuned by specifying different DCF and TDMA parameters (e.g. contention windows, slot size) in the MAClet configuration packets. The coexistence experiments can be assessed both in terms of channel activity analysis and high-level throughput performance.

Channel Activity Analysis. Figure 3 shows an example of resource repartitions between operators A and B in two consecutive beacon intervals. The figure plots the channel activity trace (in terms of RSSI values) captured by a probe built with USRP and GNU-Radio: for better distinguishing the two virtual networks, the TDMA stations

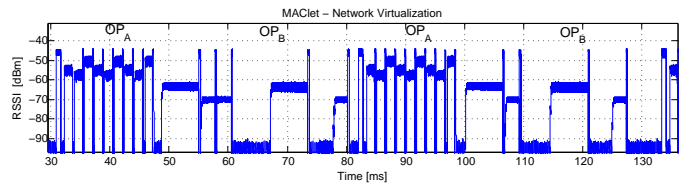


Fig. 3. An experimental trace of network virtualization with independent access schemes (TDM and DCF).

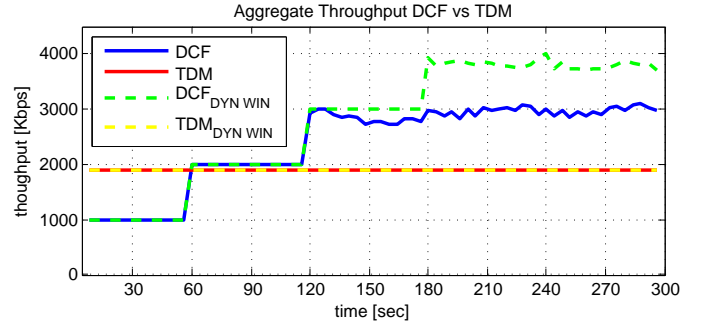


Fig. 4. Resource repartition between two different operators using different access rules (TDM and DCF).

transmit at 11 Mbps while the best effort stations transmit at lower data rates (5.5 Mbps and 2 Mbps). Note that in the first TDMA slot the channel is busy (i.e. a transmission has been originated in that slot), but no acknowledgment is received because of channel errors.

High-level Throughput. Consider a scenario where three stations access the channel by using TDMA, while five stations join sequentially the best-effort network at regular intervals of one minute. TDMA stations have a traffic rate of 640 kbps, while DCF stations work with a traffic rate of 1 Mbps. Each operator can receive an equal share of the available bandwidth (i.e. the activation time is one half of the beacon interval), or a dynamic share (e.g. the TDMA operator agrees to release the available bandwidth to the other operator).

Figure 4 shows the per-operator throughput results obtained under both the sharing policies. In case bandwidth is equally shared, the throughput of the best-effort operator (blue curve) saturates to about 3 Mbps after the third station has joined the network (note that 3Mbps is approximately one half of the total network capacity at 11 Mbps): TDMA network is obviously under utilized. By adjusting the time allocated to the best-effort operator, the third station can join the network without causing any throughput degradation. The aggregated network throughput (green line) for the best-effort network is now about 4 Mbps, while TDMA stations performance are not affected by increased DCF traffic.

REFERENCES

- [1] Y. Grunenberger, I. Tinnirello, P. Gallo, E. Goma, G. Bianchi, "Wireless card virtualization: From virtual NICs to virtual MAC machines", Future Network and Mobile Summit (FutureNetw), July 2012, Berlin.
- [2] R. Chandra and P. Bahl, "MultiNet: connecting to multiple IEEE 802.11 networks using a single wireless card", Proc. of IEEE INFOCOM 2004, Vol. 2, pp. 882-893.
- [3] FreeBSD project, <http://people.freebsd.org/~sam/BSDCan2005.pdf>.
- [4] S. Kandula, et al, "FatVAP: Aggregating AP Backhaul Capacity to Maximize Throughput", Proc. of NSDI'08, pp. 89-103, 2008.
- [5] I. Tinnirello, G. Bianchi, P. Gallo, D. Garlisi, F. Giuliano, F. Gringoli, "Wireless MAC Processors: Programming MAC Protocols on Commodity Hardware" IEEE INFOCOM, March 2012.
- [6] G. Bianchi, P. Gallo, D. Garlisi, F. Giuliano, F. Gringoli, I. Tinnirello, "MAClets: Active MAC Protocols over hard-coded devices", Proc. of ACM CONEXT 2012, Dec. 2012, Nice.