Secure IPsec based offload architectures for mobile data : architecture description and performance evaluation

Daniel Migault, Daniel Palomares, Hendrik Hendrik, Maryline Laurent

Secure IPsec based Offload Architectures for Mobile Data: Architecture Description and Performance Evaluation

Daniel Migault
Orange Lab

Daniel Palomares
Orange Lab

Hendrik
Orange Lab

Maryline Laurent Institut
Mines-TELECOM, UMR
CNRS 5157 SAMOVAR

ABSTRACT

Radio Access Network (RAN) are likely to be overloaded, and some places will not be able to provide the necessary requested bandwidth. In order to respond to the demand of bandwidth, overloaded RAN are currently offloading their traffic on WLAN. WLAN Access Points like (ISP provided xDSL boxes) are untrusted, unreliable and do not handle mobility. As a result, mobility, multihoming, and security cannot be handled by the network anymore, and must be handled by the terminal.

This paper positions offload architectures based on IPsec and shows that IPsec can provide end-to-end security, as well as seamless connectivity across IP networks. Then, the remaining of the paper evaluates how mobility on these IPsec based architectures impacts the Quality of Service (QoS) for real time applications such as an audio streaming service. QoS is measured using network interruption time and POLQA. Measurements compare TCP/HLS and UDP/RTSP over various IPsec configurations.

1. INTRODUCTION

With rising popularity of tablets and M2M applications, Cisco [6] foresees that in 2015, the average smartphone will generate a traffic of $1.3\,10^9\ bytes.month^{-1}$, which represents a 16-fold increase over the 2010 average of $79\,10^6\ bytes.month^{-1}$. In other words, aggregate smartphone traffic in 2015 will be 47 times more than in 2010.

Large part of the ISPs' revenues are provided by Services, and not facing this increasing demand on traffic represents loss of profits. As such, ISPs have to make their infrastructure ready to deal with that traffic and have three alternatives [22, 30, 38]:

- **Upgrade their infrastructure** by building the required number of cells.
- **Optimize their infrastructure** by improving the current technology and increasing each cell's capacity
- **Offload** the traffic on Alternate Networks such as WLAN.

Norman et al. [30] evaluate that offloading 52% of the traffic on indoor WiFi (resp. outdoor WiFi) reduces the costs by 4.8 (resp. 2.3) times over the optimize and upgrade scenario.

Despite a great economical advantage, offload architecture comes with an added complexity, that ISPs have to overcome so to take the full advantage provided by the offload architectures.

Firstly, one of the challenges is to compose with multiple local WLAN operators and aggregators. For example, The Cloud [45] is a Local WLAN provider, covering multiple places all over Europe, that has concluded partnerships to extend its coverage with Mobile Network Operators (MNO) such as Telenor [44], Sprint [42], AT&T [2] as well as aggregators like iPass [15].

Secondly, with multiple actors, securing the communication at the radio layer (L2) is not sufficient since the WLAN Access Point (WLAN AP) may be untrusted —i.e. does not belong to the ISP. Securing the radio layer only results in offloading end user only to the WLAN APs owned by the ISP. On the other hand, securing at the transport layer (L4) or above, at the application layer, requires that the applications are security aware, and thus must be ported to TLS [9] or DTLS [37] to be offloaded. Porting these applications to DTLS or TLS brings two issues. First developers are unlikely to modify their applications to solve an ISP issue. Second porting these applications to TLS/DTLS adds a security overhead even when they are used over a trusted network. This may impact the end user with additional certificate pop-ups, warnings, misconfiguration, additional computation or latency.

As a result, the more realistic layer to work on seems to be the IP layer (L3), which can be secured with IPsec [19]. IPsec secures the communication over untrusted WLAN APs, and untrusted networks. In addition, it provides multihoming and mobility facilities with MOBIKE [3], and MOBIKEv2 [7] used to overcome the WLAN unreliability as well as terminal mobility operations.

IPsec comes with two modes the tunnel mode and the transport mode. IPsec key management is performed by IKEv2 [17]. MOBIKE is the IKEv2 MOBility extension. Originally, it has been standardized for the tunnel mode [3], but recent work makes adoption to the transport mode with MOBIKEv2 [7]. In order to measure the advantage of using MOBIKEv2 with the transport mode, we implemented this extension on the StrongSwan [43], the reference opensource implementation of IKEv2.

MOBIKEv2 implements mobility and multihoming for both IPsec modes (the transport mode and tunnel mode) whereas MOBIKE only implements them for the tunnel mode. However, for clarity, in the remaining of the paper MOBIKE is used for the tunnel mode and MOBIKEv2 is used for the transport mode.

The use case considered in this paper is an end user with an audio streaming service. This end user is being offloaded on a WLAN network and is not anymore attached to the RAN. It goes from one WLAN AP to another with a single WLAN interface. These WLAN APs may belong to multiple different actors and thus cannot be trusted. In order to provide equivalent confidentiality on the WLAN network as on the RAN, the communication is protected at the IP layer using IPsec. The remaining questions are how to provide a QoS equivalent to the one provided by the RAN. One may consider transport layer protocol and evaluates whether dropping packet with UDP/RTSP provides any advantage over recovering lost packets using TCP/HLS. Then, one also may consider the various IPsec configurations and evaluates how MOBIKE/MOBIKEv2 improves the QoS. Similarly, one may also balance the IPsec transport mode with a reduced overhead over the tunnel mode which does not break the application communication when a mobility between WLAN APs is performed.

As a result, the offloaded communication is IPsec protected. The paper is structured as follows. Section 2 positions our work. Section 3 positions and describes different offloaded architectures: anyWLAN —that provides WLAN mobility —, the Offload Access Architecture (OAA) —that uses the IPsec tunnel mode —and the Offload Service Architecture (OSA) —that uses the IPsec transport mode. Section 4 describes our experimental platform we used to measure how moving from one WLAN AP to the other impacts the End User communication with an audio streaming service. Section 5 evaluates networking measurements for traffic captures and section 6 evaluates the QoS measured by POLQA [35] with TCP/HLS [34] and UDP/RTSP [40] protocols with various IPsec configurations. Finally, section 7 concludes our work.

In summary, we make the following contributions:

1. We describe different secure offload architectures. The described architectures differ in many aspects: The *anyWLAN* architecture is based on Layer 2 or Radio Layer security whereas the *Offload Access Architecture* (OAA) and the *Offload Service Architecture* (OSA) are based on Layer 3 or IP Layer Security (IPsec). The remaining of the paper is focused on IPsec based architectures. The main difference between *OAA* and *OSA* is that *OAA* is mostly designed to address the whole traffic —and then multiple independent services —of an end user whereas *OSA* is more likely to be specific for a dedicated service. Given the needs of an ISP, this paper enables an ISP or a MNO to chose or design a secure offload architecture.

2. We compare different ways to handle mobility with a secure IPsec communication. More precisely, MOBIKE has been designed to handle tunnel mobility, which is perfectly convenient for the *OAA*. However, *OSA* has designed for a dedicated service which makes possible end-to-end security. With end-to-end security, tunneling is not anymore mandatory, and thus, the communication can be secured with IPsec transport or tunnel mode. With IPsec transport mode IPsec mobility cannot take anymore advantage of the tunnel mobility. We thus had to design a new protocol to handle IPsec transport mode mobility: MOBIKEv2. IPsec transport mode secures a communication in a similar manner as TLS/DTLS does, even though the layers are different. For that purpose, we designed, implemented and tested this new protocol we named MOBIKEv2 [7]. Given a set of applications to offload, the paper enables to define which IPsec extension should be chosen.

3. We compare how real time applications may be configured or designed when used in conjunction of IPsec. For that purpose, we consider an audio streaming application. We measure the impact of using UDP or TCP based protocols in conjunction of the different IPsec mobility extensions. Given an offload architecture, the paper enables ISPs to chose the appropriated transport protocols for their applications.

4. Our tests considers a network approach as well as a QoS approach. Both measurements are complementary, and may provide inputs for other services than audio streaming services.

Although the abbreviation has previously been defined, we redefined these terms we refer throughout the paper:

- WLAN AP: designates the WLAN AP. end users are attached to WLAN AP at the radio layer (802.11). WLAN AP may also provide the end user and IP address.

- OAA: Offload Access Architecture, the first IPsec based architecture presented in this paper which involves IPsec tunnels and a Security Gateway.

- OSA: Offload Service Architecture, the second IPsec based architecture considered in this paper with end-to-end security, that is to say where no gateway is involved.

- HLS: HTTP Live Streaming [34]

2. RELATED WORK

Multiple papers have been proposed to enhance the QoS of the end user communication. However, most of them consider the end user connected with both its RAN interface and its WLAN. As a result, they mainly develop strategies to balance traffic between these two interfaces. Siris et al. [41], analyze how to optimize WLAN bandwidth with prediction and prefetching, Balasubramanian et al. [4] check how delaying application download over 3G networks takes advantage of WLAN networks. Similarly, Lee et al. [21] simulate from WLAN connectivity statistics how much data can be offloaded from 3G to WLAN. Our work differs from these as we do not consider the use of multiple interfaces. Our paper considers the end user is connected to a WLAN network using a single WLAN interface.

Deshpande and al. [8] investigate how to enhance connectivity among WLAN AP. They take advantage of prediction and prefetching to maximize the use of bandwidth. In fact prediction and prefetching has been explored by numerous other work. Our work differs from these as we do not consider how WLAN mobility can be enhanced by any of these mechanisms. Our work is focused on measuring how IPsec mobility extension improves the QoS of a streaming service. However, as prefetching and prediction investigated techniques, we position our architecture toward these techniques.

As far as we know few works have been made on IPsec based architecture for offload. Migault et al. have considered such architectures in conjunction with multiple interfaces with SCTP. Migault et al. [26] configures IPsec with multiple interfaces to optimize IPsec negotiation when SCTP multihoming is performed. [25] describes how SCTP can be adapted to switch from a non IPsec secured communication on the RAN to an IPsec secured communication on WLAN. [26] and [25] lead to the standardization effort [27] that details how to handle multiple interfaces with IKEv2. These papers were focused on using multiple interfaces, and this paper is concerned about mobility. More specifically, this paper considers a single interfaces, which represents the worst mobility case for a terminal with multiple interfaces.

A special attention is given to HIP [11] that provides both IPsec security and mobility facilities. HIP communications are established between crypto identifiers (Host Identity Tags or HIT), bound to an IP address. Since HITs remain fixed during the communication, IP addresses can be changed / added transparently to the application. The HIP based architecture provided by Heer et al. [13] is very close to the OSA architecture we proposed. However, HIP suffers from two drawbacks: (1) Communications are always IPsec protected and (2) HIP breaks the current IP oriented communications.

In this paper we developed MOBIKEv2 in order to take advantage of the IPsec transport mode. The main advantage of the transport mode is that it reduces the IPsec overhead. An alternative to MOBIKEv2 that also removes the tunnel overhead could be the BEET mode [28]. The BEET mode has been derived from HIP work and is a new IPsec mode. With the BEET mode, MOBIKEv2 may not be necessary and MOBIKE would be sufficient. As far as we know the tests mentioned in the paper have not been performed with the BEET mode. The reason we designed and implemented MOBIKEv2 is that MOBIKEv2 is an extension of MOBIKE and leaves IKEv2 and the kernel implementations of IPsec unchanged. On the other hand, the BEET mode requires the BEET mode to be implemented in the kernel, and IKEv2 to consider this new mode. As a result, adoption of MOBIKEv2 seems easier to be adopted than the BEET mode.

3. OFFLOAD ARCHITECTURES DESCRIPTION

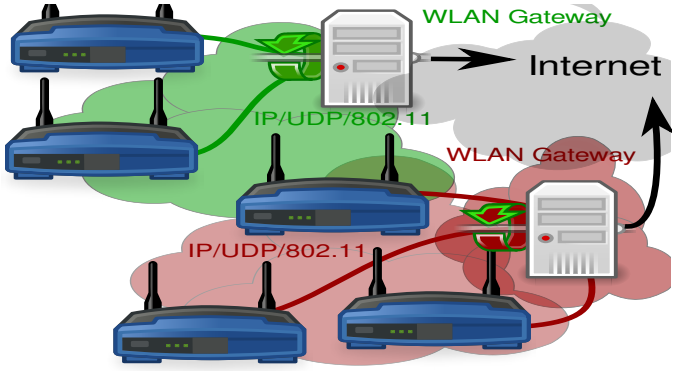
This section describes three secure offload architectures: the *anyWLAN*, the *Offload Access Architecture* (OAA) and the *Offload Service Architecture* (OSA). *anyWLAN* only considers L2 security/mobility whereas OAA and OSA considers L3 security and mobility.

Typically, an end user connected to WLAN is attached at the radio layer to WLAN AP, but the WLAN AP may also provide the IP address. Moving from one WLAN AP to another usually results in breaking both the WLAN and IP layers, which requires for both layers to proceed to a new attachment. Changing an IP address usually breaks the communication, and the application needs to be restarted. WLAN re-attachment has less impact on the applications, but still increases the interruption time of the communication. As a result, optimal offload architecture usually combines mechanisms at the radio and IP layers. Section 3.1 presents the anyWLAN architecture that defines administrative domain where the end user is able to move from one WLAN AP to the other without breaking its WLAN attachment. On the other hand, while moving across different administrative domains, WLAN and IP are updated. So to overcome these updates without breaking the communication, IP mechanisms must be involved. These mechanisms are provided by the Offload Access Architecture (OAA) in section 3.2 and the Offload Service Architecture (OSA) in section 3.3.

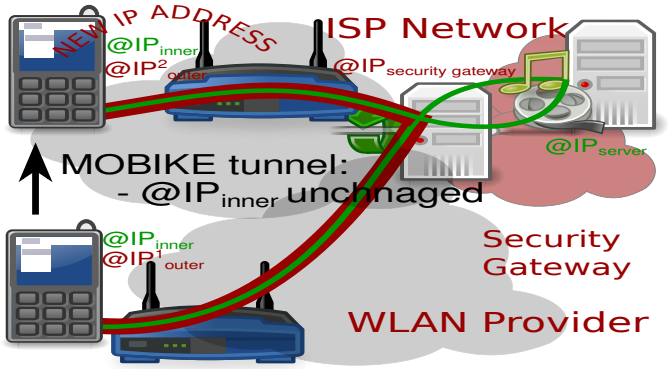
This section positions the anyWLAN, OAA and OSA architectures in terms of proposed functionalities, protocol overhead associated to each architecture, abilities to enhance the end user communication with mobility prediction and prefetching [41]. We also point how OSA and OAA can be combined with anyWLAN. To compute the protocol overhead of the different architectures, we considered Voice over IP application with G729 IP/UDP/RTP [14], as well as audio streaming protocol AAC-96Kbs over IP/RTSP and IP/HLS. Then, sections 5 and 6 experimentally measure interruption time and QoS for the audio streaming protocol. Mobility prediction and prefetching [41] are well known techniques to enhance offloaded traffic. Prediction predicts the next WLAN AP the end user will be attached to, and prefetching consists in provisioning and caching the requested content in the next WLAN AP, to optimize the use of bandwidth.

3.1 anyWLAN

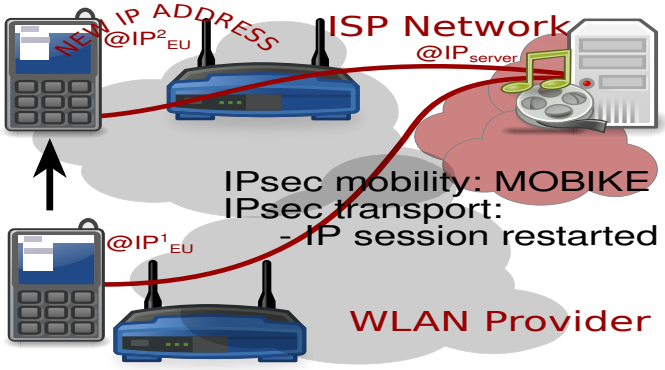
Figure 1a depicts the anyWLAN architecture. An administrative domain is defined as WLAN Gateway and all connected WLAN APs. The WLAN APs encapsulate the 802.11 traffic in a UDP/IP tunnel to the WLAN Gateway. The end



(a) anyWLAN



(b) OAA



(c) OSA

Codec	G729	AAC-96Kbs	AAC-96Kbs
Protocol	IP/UDP/RTP	IP/RTSP	IP/HLS
Packet size	60 bytes	300 bytes	1500 bytes
anyWLAN	96%	19.3%	3.86%
ESP Tunnel	96%	19.3%	3.86%
ESP transport	80%	15%	3.2%
Combination with anyWLAN with IPsec			
ESP Tunnel	203%	39%	7.86%
ESP transport	176%	35%	7%

(d) Protocol overhead (IPv4)

Figure 1: Offload Architecture Description

user is attached to the WLAN Gateway, thus WLAN attachment does not need to be re-negotiated when the end user moves from one WLAN AP to the other.

anyWLAN is secured as the WLAN attachment is not performed with the untrusted WLAN AP, but with the WLAN Gateway, located in the CORE network of the ISP or the MNOs. Path prediction and content prefetching are regular techniques to optimize bandwidth consumption in offload environments. In addition, the centralized WLAN Gateway provides built-in path prediction and prefetching. Packets lost during a mobility can be replayed using the radio layer protocol or the application TCP protocol.
AnyWLAN compared to a direct attachment to the WLAN AP adds a tunnel overhead as well as network latency between the WLAN AP and the WLAN Gateway. Compared to WLAN latency, the ADSL/cable latency is negligible. The IP/UDP/WLAN tunnel overhead is 58 bytes for IPv4 and 74 bytes for IPv6, and table 1d provides anyWLAN overhead for various real time applications. However, the main drawback of this architecture is that mobility across administrative domain is not considered, and most of the time requires the application to be restarted. Usually, restarting an application takes much longer than network attachment. As a result, ISPs or MNOs deploying anyWLAN may have large administrative domains centralized to a unique WLAN Gateway. Clustering this technology is not mature, so we believe this architecture lacks scalability. Finally end user's communications are trusted only if a trusted relation exists between the end user and the WLAN Gateway.

Finally, anyWLAN does not support mobility between administrative domains either belonging to a given ISP, or different ISPs or MNOs. As a result, we recommend to use it in combination with the OAA or OAS architecture to either avoid breaking the IP communication and to make possible a trust communication while being attached to untrusted WLAN APs.

3.2 Offload Access Architecture (OAA)

Figure 1b depicts the Offload Access Architecture somehow using the same concepts as the 3GPP IWLAN [1] architecture. The end user is attached to a WLAN AP with an IP address associated to the WLAN AP: the outer IP address. The end user gets an inner IP address from the Security Gateway. The inner IP address is used by the application and the outer IP address carries an ESP/ inner IP payload to the Security Gateway. The Security Gateway encapsulates (resp. decapsulates the inner IP payload). As a result,

when the End User changes its WLAN AP and gets a new outer IP address, the inner IP address remains unmodified, so the IP communication with the application is not broken. The IPsec tunnel is set with IKEv2 [17] which requires the MOBIKE [3] extension to update the outer IP address without breaking the communication with the inner IP address. The main advantage of OAA is that the outer IP address can be updated without breaking the communication using the inner IP address. In other words, this makes possible mobility across anyWLAN domain. More specifically, the end user can take advantage of multiple MNOs. Secondly, since secure attachment is provided at the IP layer with the Security Gateway located in the ISP CORE network, the communication remains trusted even though WLAN APs may be untrusted. Similarly to anyWLAN, OAA may easily take advantage of a centralized architecture to optimize offload bandwidth with path prediction and prefetching. In addition, Security Gateways are mature and multiple mechanisms have been designed to scale the load and clustering solutions are proposed by multiple vendors [29, 32, 33, 39]. OAA and IPsec does not provide any means to retransmit lost packets during a mobility. New IKEv2 extension may be designed for that purpose. Currently, retransmission mechanism is provided by the TCP session of the application. In our case, we used IPsec with ESP [18] using AES-128-CBC [10] with integrity check using HMAC-SHA1 [23]. The IP/IP/ESP overhead is around 64 bytes for IPv4 and 80 bytes for IPv6.

The main advantage of OAA architecture is that it provides secure mobility among WLAN APs of different administrative domains. When mobility occurs inside a single administrative domain, WLAN attachment procedure is not required. On the other hand the combination of anyWLAN and OAA provides a double encapsulation that may add latency for real time applications. Table 1d sums up the OAA ESP tunnel overhead for real time applications in combination or not with anyWLAN. Another advantage of using IPsec is that traffic can be identified by the WLAN AP using (IP addresses and SPI), and this traffic is authenticated by the Security Gateway. This simplifies billing between third party MNO that provides WLAN connectivity and the ISPs owning the Security Gateway.

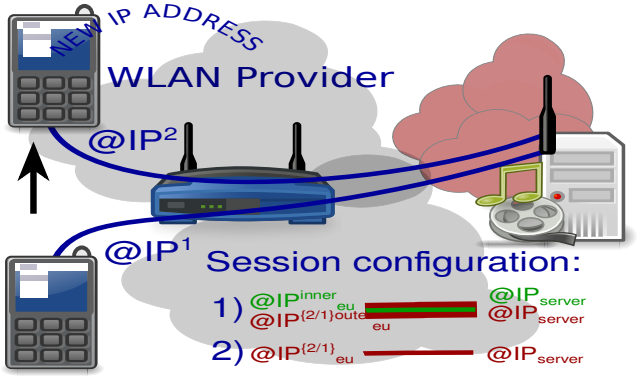
3.3 Offload Service Architecture (OSA)

Unlike OAA, designed to offload all Internet traffic of the end user to the trusted network of the ISP, OSA depicted in figure 1c, offloads a specific service. By reducing the scope of offloaded traffic, OSA is expected to ease offload infrastructure deployment by the ISPs. OSA also provides offload architectures on a per-service basis, which allows different IPsec configurations according to the specific application. For example applications that handle the change of IP address, delay tolerant applications or applications limited to a single exchange (like DNS) may remove the tunnel overhead to maximize their latency and use the IPsec transport mode. Applications that need mobility support may prefer using IPsec with the tunnel mode. In addition, OSA provides end-to-end security with no Security Gateway. This avoids traffic indirections as well as non-service specific traffic loading the Security Gateway, and overall OSA eases the management of the Quality of Service.

With end-to-end security, path prediction or prefetching can be implemented as in OAA. However, caching content does not benefit from centralizing multiple end users. As a result, this architecture is recommended for real time service that carries personal data. VoIP services are typical targets for OSA. Other services like DNS(SEC) with a last mile security are also concerned. Similarly to OAA, OSA makes secure the communication over untrusted WLAN APs. The overhead analysis is similar as the one with OAA except that the IPsec transport mode does not have a tunnel overhead. This results in a 40*bytes* bandwidth saving for both IPv4 and IPv6. Table 1d sums up how transport mode reduces the overhead. Note that with the transport mode, MOBIKEv2 [7] provides mobility for IPsec and avoids the IPsec Security Association to be re-negotiated, at the price that mobility is supported by the application.

4. EXPERIMENTAL PLATFORM DESCRIPTION

This section describes the platform depicted in figure 2a used to measure how offload mobility impacts the quality of the end user communications. We measured with a networking approach the time a mobility operation interrupts the session (i.e the switching time), and with a service approach how mobility affects the QoS. Switching time is computed from pcap traffic captures and QoS is computed by comparing the original audio stream with the recorded one, using Perceptual Objective Listening Quality Assessment POLQA [35] an ITU-T standard that covers a model to predict speech quality by means of digital speech signal analysis. For each measurement we perform around 50 tests and use the quartile representation in all figures.



(a) Architecture

	Ethernet	Indoor WiFi	Outdoor WiFi
Bandwidth (TCP)	98.9 Mbit/s	7 Mbit/s	2 Mbit/s
Latency (ms)	0.3660 ms	225.36 ms	441.28 ms
Quality	-	70/70	40/70
Signal	-	−16 dBm	−60 dBm

(b) Link Characteristics (iperf, ping)

Figure 2: Experimental Platform

The platform is made of 2 PCs DELL Latitude 5400 running

Ubuntu 12.04 for the Server and the end user, and a Linksys WRT54G for the WLAN AP. Tests are performed for an audio streaming service ACC using different bit rates 8 Kbs, 48, Kbs 96 Kbs over UDP/RTSP and HLS protocols. The different bit rates with no buffer aims at maximizing the impact of mobility on different kinds of real time applications like Voice over IP (VoIP) or Video on Demand (VoD). As we used POLQA to measure QoS, the audio stream is a made of human voice. We used VLC [46] as a client and a server.

At the network layer, we tested UDP/RTSP and HLS over the following security configurations:

- **NO_IPSEC**: defines the base line with no security and no support for mobility.
- **IPsec Tunnel NO_MOBIKE**: defines an IPsec architecture using the IPsec tunnel mode. If a change of IP address is not supported by the application or the transport layer, the application and IPsec layer must be entirely renegotiated. This configuration does not provides obvious interests as it adds a tunnel overhead on the communication. This is not balanced by the supported of mobility with MOBIKE. One possible advantage would be that the inner IP address remain unchanged during the mobility leaving the communication unchanged for the audio streams. In this case, only the outer IP addresses and the IKE SA would have been renegotiated. This is not implemented as MOBIKE provides a much efficient way to do this.
- **IPsec Transport NO_MOBIKE**: defines an IPsec architecture using the IPsec transport mode. Compared to the tunnel mode, the security overhead is reduced. If a change of IP address is not supported by the application or the transport layer, the application and IPsec layer must be entirely renegotiated.
- **IPsec Tunnel MOBIKE**: defines an IPsec architecture using the IPsec tunnel mode. In this case MOBIKE is supported which means that even if the application or the transport protocol does not support a change of IP address, the inner communication is not broken.
- **IPsec Transport MOBIKE**: defines an IPsec architecture using the IPsec transport mode. In this case MOBIKEv2 is supported which means that the IPsec layer support a change of IP address. However, if the application or the transport protocol does not support a change of IP address, the transport or application must be renegotiated, but not the IPsec. This configuration removes the tunnel overhead which may increase latency.
- **HTTPS**: defines a TLS session. This configuration is only tested with HLS. There is no support for mobility or session resumption in this case. This configuration is the TLS counter part of the IPsec Transport NO_MOBIKE configuration. It makes possible to compare the impact on QoS and network of performing security at the application/transport layer or at the IP layer.

Measurements consider different use cases with different link characteristics summed up in table 2b. The Ethernet is used for networks with high bandwidth. Indoor WiFi considers the end user is attached to a WLAN AP located in the same room. This is typically the case when end users are connected to their home DSL box with high signal quality. Outdoor WiFi considers a WLAN communication altered by the distance and multiple signal indirections —typically walls.

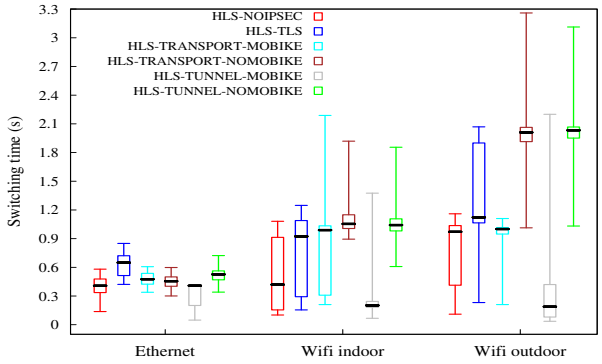
We did not consider in our tests the WLAN attachment. One reason is that we noticed 2 s delay using *wpa2_supplicant*. We suspected that WLAN drivers have not been optimized for fast attachment both in the end user and low power WLAN AP MIPS Linksys WRT54G. One reason for not optimizing the WLAN attachment procedure is that WLAN attachment has currently not considered mobility cases. In addition, WLAN attachment may be improved by multiple ways such as allowing parallel attachment of WLAN APs. This would require drivers updates of course, but would considerably affect our results and measurements. As a result, we leave optimized WLAN attachment for further studies.

5. NETWORKING MEASUREMENTS

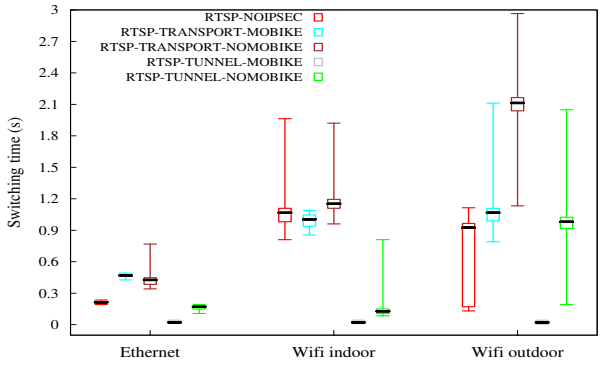
This section evaluates the networking performances while the end user moves its communication from one WLAN AP to the other. Figure 3 measures the switching time, that is the time the end user does not receive any datagram from the server, from a live capture of an audio stream of 96 Kbs.

All the configurations show that MOBIKE and IPsec with tunnel mode minimizes the switching time. In fact the use of MOBIKE avoids re-negotiating the IPsec SA. IPsec SA negotiation involves an IKEv2 4-packet exchange and in the case of EAP-SIM [12] authentication multiple additional exchanges are required. These exchanges typically delay the re-connection. In addition, the tunnel mode makes possible to update the outer IP address without updating the inner IP address seen by the application. This avoids breaking the TCP/UDP connection. The problem with re-initiating a TCP/UDP connection is that most applications have not been designed to recover from connection interruption. As a result, a lot of applications must be restarted leading to a switching time much higher than only re-establishing a TCP/UDP session. Applications that deal with session interruptions may use heartbeat messages at the application layer, and in case no response is received, the application considers checking the connectivity. Such applications hardly compete with mobility handled by the kernel as the IPsec tunnel mode with MOBIKE does. Other applications may interact more closely with the operating system, and in that case it may compete with MOBIKE and IPsec tunnel mode. However such applications require specific developments that break the layer model. Overall MOBIKE and the Tunnel mode provides a 0.3 s interruption with TCP.

With RTSP with IPsec with the tunnel mode, we suppose that UDP and IP encapsulation adds propagation delay for the audio streaming server to detect the connection is bro-



(a) HLS



(b) RTSP

Figure 3: Switching Time

ken. As a result, the server keeps on sending traffic, that is later rejected by the kernel.

In the outdoor WiFi use case, MOBIKE and IPsec transport mode provide, for both HLS and UDP a similar 1.2 s switching time as NO_IPSEC. This shows that MOBIKE makes IPsec overhead negligible, and that the switching interruption is the time needed to restart the application. In an ISP perspective, using IPsec provides security without affecting the service. However on a service provider point of view, there might be an advantage to provide mobility in addition to the security. This may be chosen on a per-application basis as explained in section 3.3. In fact, some applications do not need mobility and prefer avoiding the tunnel overhead. This is the case with DNS(SEC) for example. DNS(SEC) is a question response exchange, with relatively small payloads. When offloaded using the tunnel mode would mode then double the size of the DNS queries, without providing significant advantage. Delay Tolerant Applications are other examples where mobility is not necessary.

For HLS we also considered the case where security is provided by TLS. In the outdoor WiFi scenario, TLS provides slightly longer switching time than IPsec transport mode or the application without IPsec. The reason is that TLS requires the establishment of an additional session, and the session is handled by the application. Unlike TLS resilient connection mechanisms [20], IPsec update is triggered by the kernel, so IKEv2 exchanges proceed, in parallel of the application and most likely before the application re-initiates its session. Note also that with MOBIKE, only a single exchange is required.

Using IPsec without MOBIKE is not recommended and provides high switching time —even higher than with TLS. Again, for streaming services, the tunnel overhead is negligible. The switching time overhead mostly represents the time of the IKEv2 negotiation. Similarly to TLS, without MOBIKE, the IKEv2 is triggered by the application, i.e. when the application sends an outbound packet to a given destination. One way to mitigate this additional switching time, is that the kernel triggers the IKEv2 negotiations as soon as it gets the new IP address. This could make the IPsec SA ready when the application is restarted. Note that to provide port agility for the application, such SA should only use IP addresses as Traffic Selectors —as the source ports may be unknown. Overall, such configurations must not be considered and MOBIKE should be used instead.

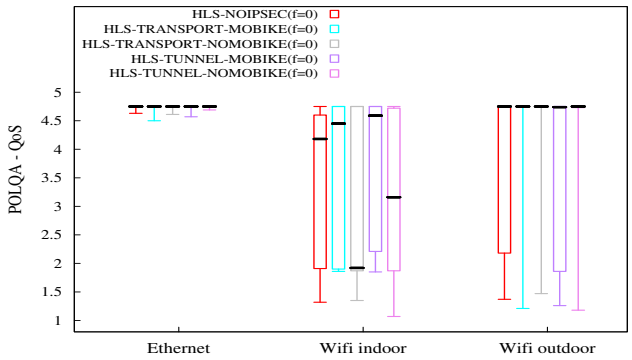
Indoor WiFi and Ethernet take advantage of low latency networks. The lower the latency is the lower the differences between the configurations is. Indoor WiFi provides the same results as the WiFi outdoor use case. On the other hand the Ethernet use case lowers most of the differences especially with HLS.

As a result, it is recommended to use IPsec with MOBIKE. Tunnel mode should be used when mobility support is required, otherwise transport mode adds very a low overhead compared to an unprotected communication.

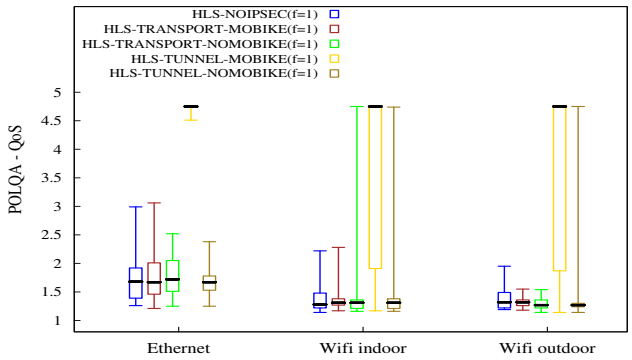
6. QOS MEASUREMENTS

Figure 4 depicts the QoS measured by POLQA and derived by comparing the original audio stream with the received stream encoded in AAC 96 Kbs. To evaluate how mobility impacts the audio stream, we measured the QoS when no mobility occurs in figure 4a with HLS and in figure 4c with RTSP. Then we measured QoS when a mobility is performed in figure 4b with HLS in figure 4d with RTSP.

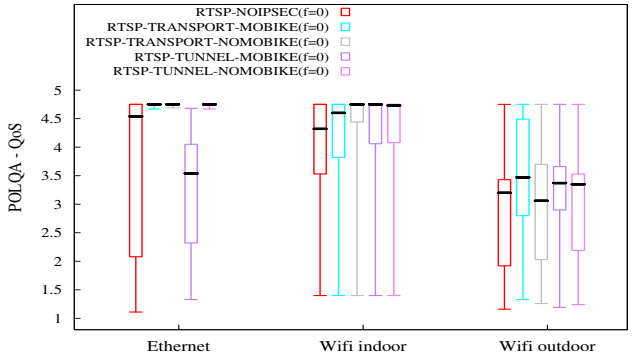
When no mobility is performed, as depicted in figures 4a and 4c, POLQA measured a better QoS with HLS than with RTSP, at least in the Ethernet and outdoor WiFi use case. HLS indoor WiFi measurements provide a lower QoS than the outdoor WiFi which seems quite unlikely given the latency difference. In addition, the indoor WiFi use case with HLS presents large variation of QoS among the configuration. As no mobility is performed, such variations are unexpected. Similarly, and especially with HLS, QoS is measured on a wide range of values ($\frac{1}{5}$ to $\frac{4.8}{5}$). In fact HLS is based on TCP, so packets are not discarded but replayed and recorded in a different speed. This results in different length in the recorded file, with blanks, and high speed replayed portions. Which portion of the speech is replayed impacts differently the QoS, which may explain the variation. POLQA sug-



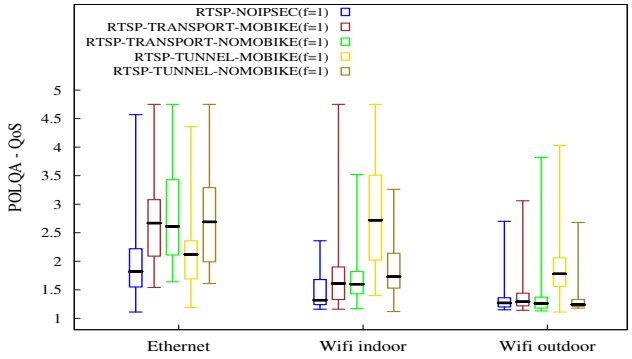
(a) TCP/HLS (no mobility)



(b) TCP/HLS (single mobility)



(c) UDP/RTSP (no mobility)



(d) UDP/RTSP (single mobility)

Figure 4: Quality of Service (POLQA)

gests that discarding packets is a better strategy to reduce the variation of the QoS. On the other hand, outdoor WiFi with RTSP has lower QoS than HLS. Finally, looking at the QoS with no mobility suggests that POLQA is very sensitive to small variations of the recorded file. Without mobility, POLQA does not clearly indicates whether RTSP or HLS should be preferred.
Results provided by POLQA should be handled carefully, it is a QoS indicator among other, and future work measuring Quality of Experience may be required.

When a mobility occurs, QoS measurements are consistent with network measurements of the switching time in section 5. More specifically, MOBIKE and the tunnel mode provides always a better QoS. This is always true for HLS, and UDP/RTSP except for the Ethernet use case as depicted in figure 4d. In fact, with Ethernet and low latency network, UDP/RTSP and IPsec configurations are expected to provide more or less the same QoS. Figure 4d shows a slight disadvantage for IPsec tunnel mode with MOBIKE, but we could not understand why QoS is lower than IPsec tunnel mode without MOBIKE.
With UDP/RTSP, the more latency is observed on the link, the lower the QoS is, even with mobility support. In fact, RTSP/UDP drops packets which directly affects the QoS, and the support of mobility with MOBIKE and the tunnel mode reduces the number of discarded packets by the application. With HLS, mobility support and packet re-ordering overcome mobility operations.

As a result, POLQA suggests to secure audio streaming services with IPsec tunnel mode and MOBIKE. The only use case where MOBIKE and IPsec tunnel does not improve significantly the QoS is for low latency networks with UDP/RTSP. When IPsec tunnel mode with MOBIKE is not used, alternate IPsec configurations have very few impact on the QoS over the NO_IPSEC configuration. POLQA also suggests to use HLS instead of UDP/RTSP. The only case UDP/RTSP seems to provide a better QoS than HLS is for low latency networks when IPsec Tunnel mode and MOBIKE cannot be used.

7. CONCLUSION

The paper shows that securing an audio streaming service using IPsec tunnel mode with the MOBIKE extension reduces the switching time and improves the QoS measured by POLQA. This shows that the lower layer network updates are performed the better it is for the application and the end user. More specifically, updating the IPsec tunnel

and leaving unchanged inner communication provides better QoS than updating IPsec and the transport layer (UDP).

The use of IPsec transport mode and MOBIKEv2 is recommended for applications that do not require mobility support or that the tunnel overhead increases significantly the QoS. Audio streaming is not one of these applications. Updating the UDP and the IPsec layer together seems to have a greater impact on the QoS then carrying the tunnel overhead and proceeding to a single IPsec tunnel update. However, applications like DNS, or Delay Tolerant Network applications or applications based on query responses with short payloads may provide the reverse.

UDP/RTSP as it uses UDP is an application that can take advantage of both IPsec transport mode and IPsec tunnel mode architectures. TCP/HLS on the other hand can hardly benefits from the IPsec transport architecture as the TCP session is broken when an IP address is updated. Measurements provided by POLQA did not show any advantage of dropping packets vs replaying the lost packets. However, we believe that further tests involving Quality of Experience would be needed, as the voice audio streams seems to us better with UDP/RTSP.

This paper considers a network analysis and QoS measurement with POLQA which provides two distinct indicators. Our recommendations concern audio streaming services, and probably can be extended to VoIP or VoD. However, we recommend that future work considers also the Quality of Experience with HLS and RTSP with IPsec transport/tunnel mode with MOBIKE.

In the network area, at least two area should be investigated more in depth. One is multiple interfaces with MPTCP [31, 36] that could result in soft handover. The second area is Network Coding [5, 16, 24] which consists in transmitting combination of packets. Combination of these two area is also of interest.

8. REFERENCES

[1] 3GPP-LTE: 3GPP system to Wireless Local Area Network (WLAN) interworking; System description, TS 23.234, Release 10. ETSI Standard, march 2011.

[2] AT&T Corporation originally American Telephone and Telegraph Company. URL: `http://www.att.com/`.

[3] M. Bakke and J. Muchow. Definitions of Managed Objects for IP Storage User Identity Authorization, may 2006. RFC 4545 (Proposed Standard).

[4] A. Balasubramanian, R. Mahajan, and A. Venkataramani. Augmenting mobile 3g using wifi. In *Proceedings of the 8th international conference on Mobile systems, applications, and services*, MobiSys '10, pages 209–222, New York, NY, USA, 2010. ACM.

[5] P. A. Chou and Y. Wu. Network Coding for the Internet and Wireless Networks. MSR-TR-2007-70. Microsoft Research, June 2007.

[6] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2010-2015. URL: `http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html`, february 2011.

[7] M. Daniel. MOBIKEv2: MOBIKE extension for Transport mode. draft. (Work in Progress) Internet Engineering Task Force, november 2014.

[8] P. Deshpande, A. Kashyap, C. Sung, and S. R. Das. Predictive methods for improved vehicular wifi access. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, MobiSys '09, pages 263–276, New York, NY, USA, 2009. ACM.

[9] T. Dierks and E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2, august 2008. RFC 5246 (Proposed Standard), Updated by RFCs 5746, 5878, 6176.

[10] S. Frankel, R. Glenn, and S. Kelly. The AES-CBC Cipher Algorithm and Its Use with IPsec, september 2003. RFC 3602 (Proposed Standard).

[11] A. Gurtov. *Host Identity Protocol (HIP): towards the secure mobile Internet*. Wiley series in communications networking & distributed systems. Wiley, 2008.

[12] H. Haverinen and J. Salowey. Extensible Authentication Protocol Method for Global System for Mobile Communications (GSM) Subscriber Identity Modules (EAP-SIM), january 2006. RFC 4186 (Informational).

[13] T. Heer, T. Jansen, R. Hummen, S. Götz, H. Wirtz, E. Weingärtner, and K. Wehrle. PiSA-SA: Municipal Wi-Fi Based on Wi-Fi Sharing. In *ICCCN*, pages 1–8, 2010.

[14] C. S. Inc. "voice and video enabled ipsec vpn (v3pn) solution reference network design". URL: `http://www.cisco.com/en/US/docs/solutions/Enterprise/WAN_and_MAN/V3PN_SRND/v3p_plan.html#wp1035443`.

[15] iPass: Enterprise Mobility Services. URL: `http://www3.ipass.com/`.

[16] S. Katti, H. Rahul, W. Hu, D. Katabi, M. MÃľdard, and J. Crowcroft. Xors in the air: practical wireless network coding. *IEEE/ACM Trans. Netw.*, 16(3):497–510, 2008.

[17] C. Kaufman, P. Hoffman, Y. Nir, and P. Eronen. Internet Key Exchange Protocol Version 2 (IKEv2), september 2010. RFC 5996 (Proposed Standard), Updated by RFC 5998.

[18] S. Kent. IP Encapsulating Security Payload (ESP), december 2005. RFC 4303 (Proposed Standard).

[19] S. Kent and K. Seo. Security Architecture for the Internet Protocol, december 2005. RFC 4301 (Proposed Standard), Updated by RFC 6040.

[20] T. Koponen, P. Eronen, and M. Särelä. Resilient connections for SSH and TLS. In *Proceedings of the annual conference on USENIX '06 Annual Technical Conference*, pages 30–30, Berkeley, CA, USA, 2006. USENIX Association.

[21] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong. Mobile data offloading: how much can wifi deliver? In *Proceedings of the 6th International COnference*, Co-NEXT '10, pages 26:1–26:12, New York, NY, USA, 2010. ACM.

[22] W. Lehr and L. W. Mcknight. Wireless Internet access: 3G vs. WiFi? *Telecommunications Policy*,

27(5-6):351–370, 2003.

[23] C. Madson and R. Glenn. The Use of HMAC-SHA-1-96 within ESP and AH, november 1998. RFC 2404 (Proposed Standard).

[24] E. Magli and P. Frossard. An overview of network coding for multimedia streaming. In *Proceedings of the 2009 IEEE international conference on Multimedia and Expo*, ICME'09, pages 1488–1491, Piscataway, NJ, USA, 2009. IEEE Press.

[25] D. Migault, D. Palomares, E. Herbert, W. You, G. G. G. Arfaoui, and M. Laurent. ISP Offload Infrastructure to minimize cost and time deployment. In *Proceedings of IEEE Global Telecommunications Conference - Communication and Information System Security (GLOBECOM '12)*, Dec. 2012.

[26] D. Migault, D. Palomares, E. Herbert, W. You, G. Ganne, G. Arfaoui, and M. Laurent. E2E: An Optimized IPsec Architecture for Secure And Fast Offload. In *International Workshop on Security of Mobile Applications IWSMA'12 (co-located with ARES'12)*, Aug. 2012.

[27] D. Migault and V. Smyslov. Clone IKE SA Extension. draft. (Work in Progress) Internet Engineering Task Force, Mar. 2014.

[28] P. Nikander and J. Melen. A Bound End-to-End Tunnel (BEET) mode for ESP. (Work in Progress), IETF, august 2008.

[29] Y. Nir. IPsec Cluster Problem Statement, october 2010. RFC 6027 (Informational).

[30] T. Norman and R. Linton. The case for Wi-Fi offload: the costs and benefits of Wi-Fi as a capacity overlay in mobile networks. Technical report, Analysys Masson, december 2011.

[31] C. Paasch, G. Detal, F. Duchene, C. Raiciu, and O. Bonaventure. Exploring mobile/wifi handover with multipath tcp. In *Proceedings of the 2012 ACM SIGCOMM workshop on Cellular networks: operations, challenges, and future design*, CellNet '12, pages 31–36, New York, NY, USA, 2012. ACM.

[32] D. Palomares, D. Migault, and M. Laurent. Failure Preventive Mechanism for IPsec Gateways. In *International Conference on Communications and Information Technology - ICCIT'13*, June 2013.

[33] D. Palomares, D. Migault, W. Velasquez, and M. Laurent. High Availability for IPsec VPN Platforms: ClusterIP Evaluation. In *The 8th International Conference on Availability, Reliability and Security - ARES2013*, Sept. 2013.

[34] R. Pantos and W. May. HTTP Live Streaming. draft. (Work in Progress) Internet Engineering Task Force, Apr. 2013.

[35] POLQA: ITU-T Recommendation P.863: Perceptual Objective listening Quality Assessment. URL: `http://www.itu.int/rec/T-REC-P.863/en`.

[36] C. Raiciu, C. Paasch, S. Barre, A. Ford, M. Honda, F. Duchene, O. Bonaventure, and M. Handley. How hard can it be? designing and implementing a deployable multipath tcp. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, NSDI'12, pages 29–29, Berkeley, CA, USA, 2012. USENIX Association.

[37] E. Rescorla and N. Modadugu. Datagram Transport Layer Security Version 1.2, january 2012. RFC 6347 (Proposed Standard).

[38] S. Risto and L. Antti. Operator's Dilemma : How to take advantage of the growing mobile Internet. URL: `http://www.notava.com/notava/uploads/Whitepapers/Internet_growth_V10.pdf`, may 2010.

[39] P. Sathyanarayan, S. Hanna, S. Melam, Y. Nir, D. Migault, and K. Pentikousis. Auto Discovery VPN Protocol. draft. (Work in Progress) Internet Engineering Task Force, July 2013.

[40] H. Schulzrinne, A. Rao, and R. Lanphier. Real Time Streaming Protocol (RTSP), april 1998. RFC 2326 (Proposed Standard).

[41] V. A. Siris and D. Kalyvas. Enhancing mobile data offloading with mobility prediction and prefetching. *CoRR*, abs/1306.3177, 2013.

[42] Sprint: Global provider of voice, data and Internet services. URL: `http://www.sprint.com/index_p.html?context=CP`.

[43] StrongSwan the OpenSource IPsec-based VPN Solution. URL: `http://www.strongswan.org`.

[44] Telenor. URL: `http://www.telenor.no/privat/`.

[45] TheCloud. URL: `http://www.thecloud.net/`.

[46] VideoLAN. URL: `http://www.videolan.org`.