IEEE Infocom 2006

A secure and performant token-based authentication for infrastructure and mesh 802.1X networks

Leonardo Maccari - maccari@lenst.det.unifi.it
Romani Fantacci - fantacci@lenst.det.unifi.it
Tommaso Pecorella - pecos@lenst.det.unifi.it

Lart - Telecommunication networks lab

www.lart.det.unifi.it

University of Florence

Introduction

- Security in wireless networks is fundamental since the lack of geographical borders has severe consequences:

Main problems

- Attackers do not need physical access
- Attackers can access layer II informations

- Mobility introduces new problems

Roaming Clients:

- Clients of the same network can be untrusted to each other
- Severe performance constraints, especially with *real time traffic*
- Mesh network model: no structure, no trust. . .

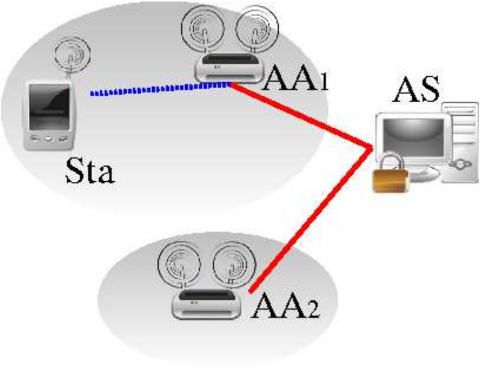
IEEE 802.1X

- IEEE 802.1X [1] has been applied to resolve some of the security protocol introduced in 802.11 standard. It has also been (indirectly) applied to other standards like 802.16. Its model is represented in figure below:

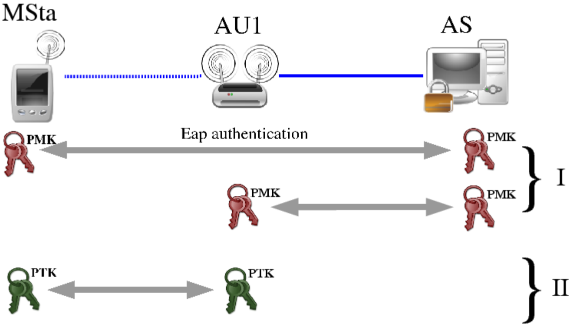
Sta: Mobile *(client)* station

AU1,AU2: Access points

AS: Authentication server



IEEE 802.1X in Infrastructure networks

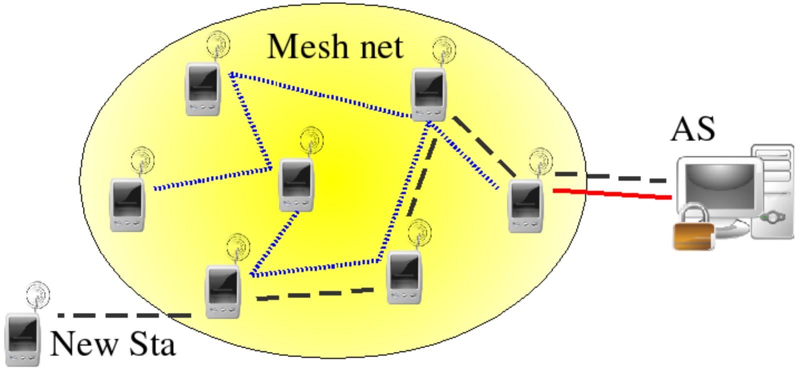


- The mobile station and the AS authenticate each other.
- Authentication involves a multi-hop path (AU1 acts as a proxy).
- Certificate-based authentication (EAP-TLS [2]) needs 7 packets!

- Two-phase authentication produces PMK (authentication) and PTK (encryption) keys

IEEE 802.1X in mesh networks

- In mesh networks each terminal behaves as an authenticator for its neighbors to reach the AS. Longer path introduces higher latency:



Summarizing

- IEEE 802.1X with EAP-TLS offers secure, mutual authentication, but introduces high latencies:
  - In mesh networks, latency due to the multi-hop path is increased by routing protocols (up to several seconds).
- If the terminals present high mobility:
  - Each handoff produces a re-authentication.
  - Frequent handoff can make the network unusable with real-time traffic.

Need for Secure, Fast handoff protocols

- On network entry, the terminal can perform a full, costly authentication.
- Then, re-use of informations generated in the first authentication can speed up following re-authentications

- Security is well defined if we define an enemy: who is our enemy?

Enemy identification

- We want our solution to be usable in Infrastructure networks but also in mesh networks where trust relationships are uncertain, the enemy model we assume is the:

Insider enemy

- A terminal that has been compromised by an external attacker, or an internal client that behaves in a malicious way, it can:
  - Cipher and decipher traffic passing through the node.
  - Inject legitimate traffic into the network.
  - Masquerade other terminals physically connected to the intruder.

- Fast Secure handoff protocols must limit the possibility of an insider enemy, we define the following guidelines for the designer:

Guideline 1 (G1)

Access control protection

- An insider enemy should not be able to introduce into the network more unauthorized terminals.

- A re-authentication is seen under the perspective of the receiving AP exactly as a new authentication. A poorly designed re-authentication scheme could let unauthorized clients to access the network.
- An insider enemy should not be able to produce valid authentication credentials to let other attackers enter the network.

Guideline 2 (G2)

Graceful degradation

- An insider enemy should not be able to obtain cryptographic keys from other terminals, if not necessary for the completion of the handoff

Explanation:

- The handoff phase implies reuse of cryptographic keys, to avoid the complete renegotiation of the keying material. To be reused old keys must be moved from one host to the other, so hosts should ask/receive keys from the neighbors of from the authentication server. A host of the network should not be put in condition to receive keys it doesn't need, this way an insider enemy cannot collect keys from the neighbors in order to decipher traffic, with an off-line attack.
- This condition is a way to avoid that compromission of a single host of the network has as a consequence compromission of the whole network, in a multi-fence secrity model [3].

Suggested Guidelines (SG)

- We introduce two more guidelines, as a generic suggestion for enhancing the security level of the protocol.

Limit the impact of Denial Of Service attacks:

- A re-authentication protocol should be designed in a way that no new denial of service are introduced, apart from the ones already existent, even for an internal attacker.
- New denial of services could be introduced if trust relationships are not tightly defined.

Centralized Management

- Even if used in mesh p2p networks, having some centralized authority that monitors the state of the network (such as movements of the clients), is of great help to the network manager.
- Making this authority participate to the handoff phase is different from simple signaling. It can lead to longer delays but offers more protection.

A token-based re-authentication scheme

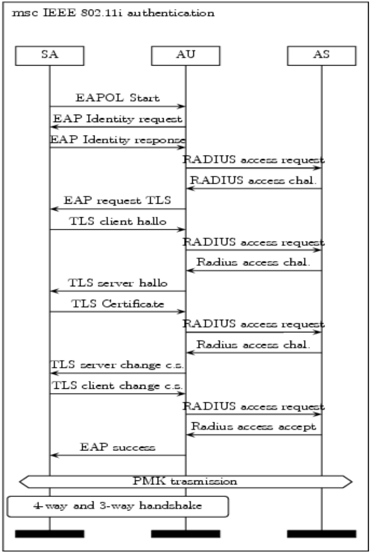
- Our solution is based on a 2-way handshake between the host that is performing the handoff and the authentication server (*AS*).
- We decided to involve the *AS* in every handoff:
  - + it is easier to maintain access control
  - + we have a centralized entity monitoring the network
  - - every handoff implies a multi-hop packet exchange. We limited the exchange to the minimum possible (2 packets).
- Our solution makes use of authentication *tokens*, i.e. keying material that an authenticator of the network (AP for infrastructure network, whatever host for a mesh network) must provide to the AS to obtain the *PMK* key.

The exchange:

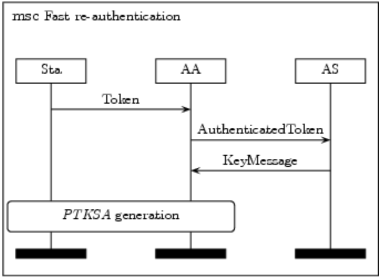
- Upon first entry in the network the client *Sta* performs a full EAP-TLS authentication. This authentication generates a *PMK* key, as depicted in fig 3, this key remains in AP1.
- Whenever *Sta* is performing an handoff (i.e. moving from AP1 to AP2) AP2 should receive the *PMK* key, to avoid a full re-authentication.
- AP2 can receive *PMK* only from the authentication server, so it must issue a request to the *AS* . . .
- . . . but G2 tells us that AP2 should not be able to obtain just any key it requests.
- AP2 should add to the request some cryptographic material (a *TOKEN*) to proof that it is in contact with some host (*Sta*) that owns the *PMK* key.
- So when *Sta* is performing the handoff, it forges a *TOKEN* based on the *PMK* it owns and passes it to AP2, AP2 issues a signed request including the *TOKEN* to the *AS*, the *AS* verifies the *TOKEN* and sends AP2 the requested key.

Key handshakes:

EAP-TLS



Fast re-authentication



Note that a 7-way handshake has been compressed into a 2-way handshake, using Identity field to carry the *TOKEN*.

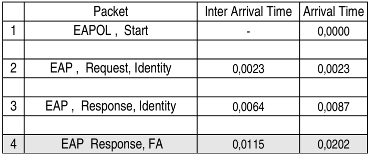
Details of 802.11 implementation:

- When the 802.1X authentication ends both authenticating parties share a *PMK* key, and they can use it to perform re-authentications.
- The Token is a signed with the *PMK* key and forwarded from the AP to the *AS*, it includes some identifier of the AP (essid).
- The path from the AP to the *AS* is a RADIUS tunnel, so that the token arrives to the *AS* with proof of authentication of the AP. The identifier included in the token must correspond with the RADIUS key used, and known to the *AS*.
- With last message, the *PMK* is moved to the AP.
- With modifications, the authenticating key used could be the *AMSK* key, from TLS protocol.*AMSK* is not exported by the endpoints as *PMK*, so it can be used to generate new *PMK* for each handoff, thus increasing security level, but also increasing complexity.

Details of real testbed implementation:

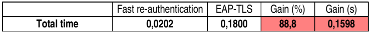
The *fast re-authentication* solution has been implemented in an infrastructure testbed [4], the link between the two AP and the *AS* is one single hop with static routing fig 3. 7 successful re-authentication had been performed to make measurements.

Fast Re-authentication inter arrival times



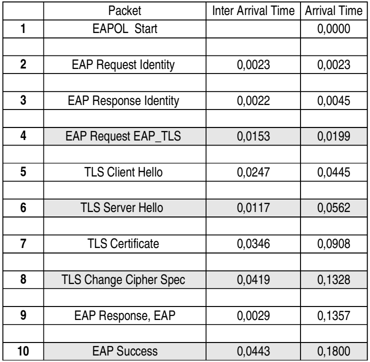
	Packet	Inter Arrival Time	Arrival Time
1	EAPOL , Start	-	0,0000
2	EAP , Request, Identity	0,0023	0,0023
3	EAP , Response, Identity	0,0064	0,0087
4	EAP Response, FA	0,0115	0,0202

Performance comparison



	Fast re-authentication	EAP-TLS	Gain (%)	Gain (s)
Total time	**0,0202**	**0,1800**	**88,8**	**0,1598**

EAP-TLS inter arrival times

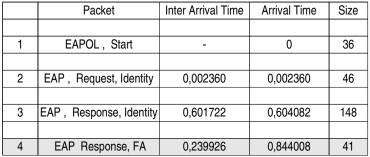


	Packet	Inter Arrival Time	Arrival Time
1	EAPOL Start		0,0000
2	EAP Request Identity	0,0023	0,0023
3	EAP Response Identity	0,0022	0,0045
4	EAP Request EAP_TLS	0,0153	0,0199
5	TLS Client Hello	0,0247	0,0445
6	TLS Server Hello	0,0117	0,0562
7	TLS Certificate	0,0346	0,0908
8	TLS Change Cipher Spec	0,0419	0,1328
9	EAP Response, EAP	0,0029	0,1357
10	EAP Success	0,0443	0,1800

Details of real testbed implementation:

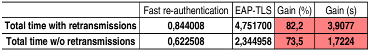
The *fast re-authentication* solution has been implemented also in a meshAP network a mesh network of access points, each one having an infrastructure subnet. Inter access point routing was performed with OLSR [5] protocol.

Fast Re-authentication inter arrival times



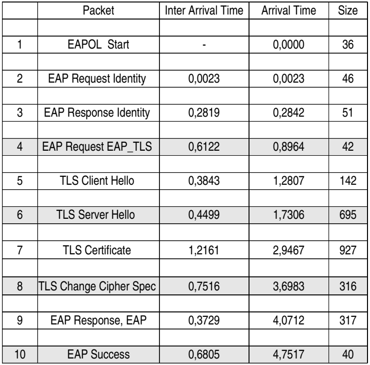
	Packet	Inter Arrival Time	Arrival Time	Size
1	EAPOL , Start	-	0	36
2	EAP , Request, Identity	0,002360	0,002360	46
3	EAP , Response, Identity	0,601722	0,604082	148
4	EAP Response, FA	0,239926	0,844008	41

Performance comparison



	Fast re-authentication	EAP-TLS	Gain (%)	Gain (s)
Total time with retransmissions	**0,844008**	**4,751700**	**82,2**	**3,9077**
Total time w/o retransmissions	**0,622508**	**2,344958**	**73,5**	**1,7224**

EAP-TLS inter arrival times



	Packet	Inter Arrival Time	Arrival Time	Size
1	EAPOL Start	-	0,0000	36
2	EAP Request Identity	0,0023	0,0023	46
3	EAP Response Identity	0,2819	0,2842	51
4	EAP Request EAP_TLS	0,6122	0,8964	42
5	TLS Client Hello	0,3843	1,2807	142
6	TLS Server Hello	0,4499	1,7306	695
7	TLS Certificate	1,2161	2,9467	927
8	TLS Change Cipher Spec	0,7516	3,6983	316
9	EAP Response, EAP	0,3729	4,0712	317
10	EAP Success	0,6805	4,7517	40

Details of real meshAP implementation:

- Each AP was equipped with 2 NIC, one for its subnet and one for the backbone network (prism2 chipset with hostap software).
- 50 handoffs had been performed, 25 of those presented retransmission of RADIUS packets over the backbone.
- Inter arrival time are calculated under a client point of view, grayed out packets in the tables represent packets traversing the whole backbone network back and forth.
- Path from AP to *AS* was 3 hops.

Conclusions:

- *fast re-authentication* has been performing better then standard EAP-TLS with gain ranging from 73% to 88%, with total time under 0.85 seconds even in meshAP environment, greatly reduces latency for real time communications
- in mesh environments collision of packets lead to timeouted retransmissions (3 seconds for RADIUS protocol). Multiple retransmission into the same session lead to peaks over 12 seconds for re-authentication. *fast re-authentication*, reducing long path packets reduces also probability of retransmissions
- *fast re-authentication* respects security guidelines defined and permits easy management of the network, always involving the *AS* into the re-authentication.

- Institute of Electrical and Electronic Engineers, Inc., *IEEE Standard for Local and metropolitan area networks Port-Based Network Access Control*, IEEE Std., 2001.
- B. Aboba and D. Simon, “Ppp eap tls authentication protocol,” RFC 2716, 1999.
- Y. Hao, L. Haiyun, Y. Fan, L. Songwu, and Z. Lixia, “Security in mobile ad hoc networks: Challenges and solutions,” *Wireless Communications, IEEE*, vol. 11, pp. 38 – 47, 2004.
- L. Maccari, R. Fantacci, T. Pecorella, and F. Frosali, “Secure, fast handhoff techniques for 802.1x based wireless network,” in *Communications, 2006 IEEE International Conference*, 2006.
- T. Clausen and P. Jacquet, “Optimized link state routing protocol (olsr),” RFC 3626, 2003.