Certificate Validation towards Efficient Trust Establishment in Mobile Ad Hoc Networks

Konstantinos Papapanagiotou[1]

Dept. of Informatics and Telecommunications
University of Athens
conpap@di.uoa.gr

Abstract. This thesis studies trust establishment in wireless, ad hoc networks. ADOPT is presented, a complete scheme for certificate validation in wireless, ad hoc networks. Its architecture is based on a distributed version of OCSP and more specifically on OCSP response caching in carefully selected nodes of the network. The method for locating cached OCSP response is thoroughly analyzed, along with techniques for caching optimization and especially efficient methods for choosing caching nodes as well as cache update and deletion policies. ADOPT is evaluated regarding its efficiency through carefully selected simulation scenarios. Moreover, the ADOPT scheme is adjusted in order to function efficiently in vehicular ad hoc networks. Simulation studies were performed to evaluate ADOPT in such an environment. In order to improve ADOPT's robustness and resilience to possible attacks, it is combined with a trust establishment scheme, namely ATF. Simulation tests prove that trust integration in ADOPT not only increases the security of the scheme but also its overall performance. Finally, ADOPT is compared with other, similar certificate validation schemes for wireless, ad hoc networks. Emphasis is given on its optimal performance concerning the rapid location of fresh certificate status information and the reduced overhead introduced to both the network and its nodes

1 Introduction

Public key cryptography and digital certificates came about along with the need for a means to revoke keys and certificates respectively. Most frequent reasons for revocation would be key loss or violation of terms of agreement. X.509 certificate format was one of the first standards to appear in order to specify the form of a digital certificate. Nowadays, it is considered the most widely used format. It introduced the concept of CRLs as a means for revocation. Such lists would be published periodically by the Certification Authority, the same entity that publishes certificates, and would contain information about the certificates that had been revoked until the time of its issuance. This CRL would then either be distributed to interested parties, or retrieved by them, scenarios that are referred to as CRL pushing and pulling

[1] Dissertation advisor: Panagiotis Georgiadis, Professor

respectively. The major drawbacks of this method are the potentially increased size of the lists, and the freshness of CSI which is included. Actually, practice showed that CRLs may grow significantly with time, reaching a size of several megabytes. In addition CSI freshness depends on the CRL issuance period: frequent CRLs increase communication overhead while infrequent CRLs may introduce a security risk due to outdated status information.

2 Main Results

The need for an "online" method that would provide current CSI was apparent since the early days of X.509 certificates. In the IETF OCSP was proposed, a simple online, request-response protocol that could be able to provide accurate, up to date CSI. In an OCSP request, a client references a certificate requesting its status. This request is forwarded to a server, so called OCSP Responder that usually has a direct link to the CA's certificate repository. Thus, the responder has always access to current revocation information. Ideally, each CA can provide its own OCSP service, while OCSP responders can also be interconnected in order to forward to each other requests that cannot be replied. However, in real world implementations OCSP Responders frequently depend on information gathered from CRLs, and, thus, the issued response would be as fresh as the corresponding CRL. Furthermore, the specification demands that OCSP clients should construct a full certificate path for the certificate they wish to validate. This task may be trivial for most cases, but requires increased processing power in order to validate multiple digital signatures. Actually, the revocation status for each certificate in the certification path should be also checked.

Most of the efforts to provide an efficient certificate validation protocol in mobile and wireless environments were mainly based on the aforementioned protocols. For instance, the Open Mobile Alliance has issued a Mobile Profile for OCSP. The standardising process of this profile was initially introduced by the WAP Forum in order to provide certificate services to GSM mobile phones. In this specification the OMA observes that OCSP messages can potentially become very large and complex. Thus, their specification mostly sets limitations to OCSP in order to assure that OCSP requests and responses have a limited size. In detail, they restrict the number of queried certificates in a request to one. It is expected that an OCSP response, as specified by OMA will be less than 3000 bytes. Furthermore, attention is drawn to response freshness. Clients may use nonces and are advised not to accept outdated responses, or responses that indicate that a newer response should exist. If they do not get a response within a certain time period should retry by sending another request message. Time parameters are specified according to GSM specifications. Finally, response caching is mentioned, solely for future use by the client itself.

Within IETF a lightweight OCSP profile was also proposed, designed for minimizing communications bandwidth and client-side processing. This specification supports response caching and an effort is made to minimize OCSP messages' size. As in limitations are set so that each OCSP request contains only one queried certificate. However, a response may contain CSI for additional certificates in order to facilitate

caching. Most message extension fields are not supported. Nonces are also supported, however it is suggested that freshness is assured through the use of an accurate source of time and efficient clock synchronization. Finally, the authorityInfoAccess extension of X.509 certificates is mentioned as the sole method for OCSP Responder Service Discovery.

Clearly, existing standardization activities keep up with traditional networking standards, by adapting existing protocols rather than introducing new ones. In this fashion, limitations are set in OCSP in order to adapt to new conditions, while only SCVP is a completely new proposal. Furthermore, we observe that IETF specifications are highly technical, offering exact protocol specifications and procedures. Message fields and restrictions are carefully and strictly set, while guidelines for handling such messages are given for both client and server implementations. On the other hand, little information is provided regarding certificate validation service discovery, expected performance in practice, security and performance tradeoffs and more. The OMA standard has the same characteristics in a more concise format.

Apart from efforts made by standardization bodies, various certificate validation protocols have been proposed to be deployed in wireless and mobile environments. In this section we briefly present some of these proposals. The Certificate Revocation System (CRS), also referred to as Novomodo offers an efficient solution when processing and bandwidth constrains exist. When creating a certificate, CRS proposes the usage two additional fields, Y and X, which are unique per certificate. These are calculated using a hash function over some secret values, Y0 and X0 respectively, initially chosen by the CA. When a relying party asks for validation of a certificate, the CA responds with a hashing value of Y', when valid, or X' when invalid, using private keys Y0 and X0, respectively. Only the CA can calculate these values, whilst any relying party can verify them. Thus, Novomodo provides an efficient way for validating certificates using 20-byte messages while its security is practically based on hash function collision resistance. One major disadvantage is the periodicity of validation proofs. For example, when a validation proof is provided for each day of the year, 365 hashes are required in total. Even if this period is reduced to one hour, the problem remains, and the total amount of required hashing dramatically increases.

CPC-OCSP (Client Partially Cached OCSP) is a protocol that aims to optimize OCSP for use in wireless environments and especially in m-commerce applications. In particular, each node can cache responses that it receives in order to use them later. Furthermore, CPC-OCSP introduces a novel way of updating cached responses in order to prolong their validity. This is achieved by the use of two hash parameters. A CPC-OCSP responder adds a parameter R in each response that it issues. Moreover, it specifies a parameter d, which corresponds to the maximum time a response can be kept in cache. The R parameter is calculated by d hashes of a random value R0. A node receiving a CPC-OCSP response with a parameter R in time t can keep it in its cache for future use. If this node wanted to check the same certificate's status in time t'>t using traditional OCSP it would have to issue a new OCSP request. In CPC-OCSP the cached response can be updated by acquiring a value R' (Type A response). R' is computed by the responder using the equation: R'=hd-1(R0). The node that receives R' can verify it by using the equation: R=h(R'). When the time that corresponds to d passes, all nodes that have cached responses must request new ones

(Type B response). Alternatively, if a certificate is revoked, a new R parameter is computed (Type C response). The MBS-OCSP (Merkle Based Server OCSP) modifies CPC-OCSP using Merkle hash trees to avoid pre-agreement of the parameter d. Finally, schemes that support segmented CRLs have been considered for P2P networks.

ADOPT is an on-demand, distributed OCSP scheme, based on cached OCSP responses, and designed in such a way, so that it can be successfully and efficiently deployed in MANETs. Its purpose is to create a fast, light, distributed and always-available certificate revocation protocol for MANETs.

OCSP is a simple protocol involving requests and responses that provide the current status of one or more certificates. A client can send a request to a server (usually called OCSP Responder) asking for information on the status of one or more certificates. This request can be digitally signed and contains a reference to the queried certificate(s) (certID). The server responds with a signed message that contains the status of the referenced certificate(s). The response message also contains time and date information. OCSP responses are always digitally signed either by the CA, a trusted or an authorized responder.

We distinguish three different kinds of nodes in ADOPT: ServerNodes, CachingNodes and ClientNodes. ClientNodes request the status of a certificate by broadcasting a message similar to an OCSP request. CachingNodes cache pre-issued and pre-signed OCSP responses and act as OCSP responders by providing such responses when needed. ServerNodes are nodes that announce the revocation status of the certificates, such as OCSP responders. They issue and sign certificate status responses which are then stored in CachingNodes. A ClientNode wishing to determine the status of a certificate forms an OCSP-like request message. In traditional OCSP, this message should then be sent to an OCSP responder, which would be identified by the authorityInfoAccess extension of the X.509 certificate. However, MANETs are highly dynamic in nature, as nodes may enter or leave the network anytime and may be moving constantly. Therefore, a DSR-like mechanism has been proposed, more appropriate for such environments. Thus, the request message is broadcasted by the ClientNode. Intermediate nodes that receive the message re-broadcast it if they do not act as CachingNodes. On the other hand, CachingNodes examine their cache for a pre-issued response corresponding to the requested certificate. If such a response is found, it is forwarded back to the ClientNode; else the request message is re-broadcasted. Similarly, if the message reaches a ServerNode, a corresponding response is issued.

Clearly, a request message may be circulating in the MANET without ever getting a corresponding response. To avoid this problem, ADOPT proposes a solution similar to the one proposed for resolving routing loops in DSR. In detail, the ClientNode determines the maximum number of hops that the request message is allowed to travel through. This TTL (Time-To-Live) parameter is included in the request message. Every intermediate node that receives the message decreases TTL by one, until the maximum number of hops is reached and the message is dropped.

The main advantage of ADOPT lies on the fact that the nodes of a MANET can receive up-to-date CSI anytime, using a distributed scheme that ensures the availability of this service. In addition, this information is delivered with a minimum cost in terms of both network and node resources. As a matter of fact, OCSP

messages are rather small in size and CachingNodes do not need to re-sign cached information status. The authenticity and integrity of the responses can be verified by the OCSP responder's signature on the response. However, the freshness of CSI depends on the freshness of the cached responses and thus, on the mechanism used for cache updating.

OCSP requests reference the queried certificate using a hash of the issuer's name and key as well as the serial number of the certificate. These fields uniquely identify a certificate. OCSP responses include three time parameters, critical to OCSP's operation. The first one, indicated by the field producedAt, denotes the time when the OCSP response was issued. Two additional parameters specify the validity interval of the OCSP response. In detail, thisUpdate indicates when revocation information regarding the queried certificate was last obtained, while nextUpdate is the time when the responder is expected to have new information concerning this certificate.

ARes freshness can be of a great importance to ClientNodes as some critical user applications may require up-to-date responses. ADOPT introduces a parameter (updateTime) in the request message that allows a ClientNode to specify how fresh the expected response should be. In such terms, if a CachingNode does not have a fresh enough response, it re-broadcasts or drops the request, depending on the TTL parameter.

Ideally, a CachingNode should deliver the most recently issued cached response. Nevertheless, it is possible that its cache is not updated. An efficient mechanism for cache updating should be in place to ensure that CachingNodes get the most updated responses. ADOPT suggests that CachingNodes get updated directly from an OCSP responder (ServerNode), either periodically or on demand. Even when communication with ServerNodes occurs using out-of-band means (e.g., through a GSM or GPRS bridge), it is possible that these OCSP responders may not be always available. Thus, ADOPT also suggests that CachingNodes can eavesdrop on messages that they forward in order to detect ARes designated for other nodes but also useful to them, for updating their cache.

Efficient cache placement policies ensure that there is no unwanted flooding of OCSP re-quests in the MANET. Cached ARes may be placed in strategic elements within a MANET, for example in high mobility nodes. Each node in the network should be able to reach a CachingNode with a cached response to his request in a few hops, so that AReq do not have to travel far in the network. Moreover, each node chooses for itself a cache update and deletion policy. Its decision depends on its position within the MANET its resources in terms of processing power, memory capacity and power autonomy. Thus, a node may choose between a greedy, selective and no-caching policy. In ADOPT, the candidate states of a CachingNode are:

- Greedy Caching State. The node caches every ARes that passes through it.
- Selective Caching State. The node caches a response after m appearances, with m being the popularity index of that response.

Overall, the caching strategies ensure that network resources are wisely spent in legitimate protocol runs, initiated by good-willing entities. Some corresponding time thresholds have been proposed in. ADOPT's TTL and Waiting Window (WW) parameters ensure that request propagation stops once a response has been located. A ClientNode has to set the TTL parameter, which specifies the maximum number of nodes that the request can pass through. If a response is not found within the specified

number of nodes, the request will be dropped. The ClientNode will be able to resend the same request with a different TTL parameter, depending on the WW. The Waiting Window parameter indicates the time a ClientNode has to wait until receiving an ARes and its calculation is based on a node's observations of network delays. Evidently, a legitimate request message will only reach a specific number of hosts, without forming any loops, thus consuming only the necessary network resources.

The presence of malicious or selfish nodes is substantial, and should be considered when providing CSI. In this section we analyze how selfishness and malicious behaviour can be dealt with using ATF as a trust component. However, we do not take into account selfish or malicious behaviours against the robustness of other protocol layers, such as Route Request flooding, or routing table fabrication, materialized in the network layer of a MANET, since we assume that these attacks will be prevented on the corresponding layer of the stack. We only discuss selfishness and malicious behaviour on the ADOPT layer. We first consider a malicious node that either initiates flooding attacks to cause serious disruption to ADOPT, or fabricates the valid status information of a certificate.

- AReq flooding. A malicious node starts flooding the network with an invalid AReq, i.e., asking for the status of a certificate that does not exist..
- ARes flooding. This similar type of attack might cause more damage as far as robustness is concerned.
- ARes Fabrication. Malicious caching nodes can fabricate responses in order to trick ClientNodes into acting as if they were valid.

Apart from being malicious, some nodes may be selfish as well. Such nodes do not wish to spend their resources for the profit of other peers, or even for the social welfare. Additionally, they demand from others to use resources for their own profit. In terms of ADOPT, selfish nodes may decide to always follow a non-caching policy. Alternatively, selfish nodes may choose not to process request messages at all. In the latter case, they relay the AReq messages without checking their cache for responses or reducing the TTL. Thus, a request message will travel in the MANET more than intended. Finally, selfish nodes may decide not to forward any request or response message. In any one of the aforementioned cases, such nodes affect the performance of ADOPT or may even cause disruption of the service.

The attacks analyzed are rather simple but if deployed on a large scale in a MANET, they could result in network congestion and partitioning as well as significant node resource consumption. In order to prevent and deal with these attacks we propose the use of the ATF framework as a trust component. Such a component would evaluate nodes' trustworthiness according to their behaviour so that peers could decide whether they should trust each other. In this section we examine how the use of ATF can enhance ADOPT's efficiency and prevent the aforementioned attacks. We also propose some customizations for ATF in order to match ADOPT's needs.

We consider here a MANET in which ATF is already deployed and Trust Values are estimated for certain node functions (e.g., packet forwarding). Even though a generic trust framework which estimates nodes' trust based on their performance on fundamental functions would be useful for any MANET application, the calculation of application-specific TVs would increase the efficiency of the application and of the network overall. ATF supports any trust-aware function and application, and, thus can also support ADOPT. A node receiving an AReq or ARes can retrieve the TVs of the

node that issued these messages from its Trust Matrix. Then, a decision should be made, based on the retrieved TVs, whether the originator should be trusted or not. In case of an AReq, the message should be processed as normal if the originator is considered trusted and, thus, the intermediate node should seek for a cached response in his cache and act accordingly. Similarly, an ARes should be forwarded to the next node if the originator is considered trusted. Conversely, if the node which issued the corresponding message should not be trusted, the request or response should be dropped. Using this approach AReq and ARes flooding attacks can be successfully mitigated. A legitimate node is expected to soon collect enough TVs in order to characterise a node as malicious. Hence, it will ignore and drop any requests or responses that originate from such nodes. Furthermore, in section 4.4 we introduce optimisations in order to facilitate the detection and isolation of misbehaving nodes. Thus, the ARes fabrication attack is also faced, as nodes that continuously fabricate invalid responses will be eventually isolated.

In a prototype simulation implementation of the ADOPT and ATF schemes, we have evaluated the performance of the integrated scheme, using the J-SIM wireless package simulator. Simulation results show that it is preferable to pay an overhead in communication cost for detecting and isolating malicious recommenders and formulating trusted paths via the ATF procedures, than to propagate and process fabricated CSI responses. ADOPT can thus rapidly locate legitimate CSI. This information is transferred through trusted paths, established by the nodes when using ATF mechanisms. Processing overhead is also minimized, since ClientNodes do not need to evaluate responses that ATF identifies as having originated from, or forwarded through malicious nodes.

As we have already seen, a variety of certificate validation protocols exists, each one of them demonstrating a different set of attributes. The described schemes can be applied in MANETs, even though they are based on protocols that were designed for fixed networks. However, MANETs introduce specific demands. Efficient certificate validation protocols for MANETs should bear characteristics such as: low bandwidth usage, little processing requirements, fast location of fresh CSI even without the presence of a centralized server and more. In this section we present some important evaluation criteria that should be taken into account when selecting a validation scheme for MANETs. Furthermore, we demonstrate through a proof of concept implementation significant performance parameters of different protocols.

The criteria that will be used in order to evaluate certificate validation schemes in MANETs should greatly depend on their dynamicity. MANET nodes are not always available as they may join and leave the network at various times. Usually, they have limited processing power, as well as power and memory capacity. Furthermore, different applications that can be deployed in MANETs may have different certificate validation requirements: some may require the freshest available CSI while some others may be willing to settle with older CSI in order to get a response as soon as possible. Hence, the selected criteria should depict all the requirements that both the type of the network and the applications that are deployed in it impose. Here, we take on three evaluation domains, namely security, management and performance, adjust them in order to apply them in a MANET environment and add some criteria that should be considered in our case.

Naturally, the most important aspect of such schemes is security. The requirement for CSI authentication, integrity and availability is vital, as a certificate validation scheme for MANETs should not assume a secure underlying protocol such as TLS. Transparency, an important management criterion, dictates that a validation scheme should function regardless of any underlying protocols. Such an overlay protocol can operate on top of any other routing, or application protocol, providing secure CSI in a self-efficient manner. Furthermore, it should be resistant to attacks by malicious nodes that, for example, can alter or drop validation messages. In the first case, a MANET node may be forced to verify digital signatures of false messages. Of course, such a signature will be proved invalid but the node will have consumed significant amount of processing power and thus, energy. In the second case, malicious users can decide not to forward validation messages in a manner that nodes may never receive accurate and fresh CSI. Similarly, selfish nodes may drop an amount of messages. Efficient certificate validation schemes for MANETs should be resistant to such behavior. Here, we distinguish mainly two important security sets: the first includes the provision of authentication, integrity and availability and the second deals with resistance to various kinds of malicious attacks or selfishness.

Efficient management is another important requirement. We have already mentioned the need for transparency, from a security point of view. Furthermore, it is evident that the use of a validation protocol should not require any knowledge of its internal functionality. The location, retrieval and verification of CSI must be automated. Such protocols should be kept simple in order to facilitate management. Environment restrictions, posed by MANETs' requirements have to be taken into account. For instance, MANET nodes are not always present. Therefore, CSI should be distributed and cached in various nodes in order to provide constant availability of validation information. Availability, as we previously mentioned, is a classic security requirement. In this case, we examine availability as a management issue according to which a validation mechanism has to ensure that CSI is efficiently distributed in a MANET so that it remains available even without the presence of some nodes. Thus, management evaluation criteria are namely transparency, complexity and CSI distribution and availability.

In terms of performance we emphasize on cost. Cost can have various aspects such as computational cost, communication cost and more. As we have already pointed out, the majority of MANET nodes is expected to have minimal processing capacity, while the wireless communication medium is also restricted. Consequently, a certificate validation scheme functioning as an overlay protocol should introduce the least possible overhead, both in terms of the size of messages that are circulating in the network and of the processing power that is required to process such messages. Furthermore, nodes' storage capacity is also limited. Thus, information that should be stored for certificate validation should not occupy a significant amount of space. An additional important requirement for certificate validation protocols is CSI freshness. Freshness is a critical parameter both in terms of security and performance. Out of date CSI could possibly lead to impersonation attacks that could compromise the security of the MANET. An efficient validation protocol should be capable to locate fresh, if possibly the freshest available, CSI in a short period of time without introducing significant overhead. Scalability is an equally important issue for any scheme designed to operate in a MANET environment. The size of a MANET in

terms of nodes may vary it time. An efficient protocol should operate effectively both in dense and sparse topologies

The aforementioned evaluation criteria comprise a solid evaluation framework for certificate validation protocols that are designed to operate in MANET environments. We have examined three distinctive validation protocols using a practical proof of concept implementation: pulling d-CRLs, CPC-OCSP and ADOPT. In order to implement CRL pulling, we created small request messages that a node broadcasts when it wishes to acquire a new CRL. A CRL scheme based on pushing would be inappropriate for use in MANETs as periodically broadcasting CRLs to all nodes would introduce a significant overhead. Furthermore, segmented CRLs cannot be implemented in dynamic environments as segments would not be stable. CPC-OCSP was chosen over MBS-OCSP as an OCSP-based scheme that can be deployed in MANETs, mainly because computations involving Merkle hash trees require additional processing power. Furthermore, SCVP is not considered efficient for MANET use, primarily because it introduces unnecessary overhead through the use of the CMS format. In addition, DPV is usually unnecessary in MANET environments as certificates are in most cases issued by a single trusted CA or the certificate chain can be stored in each node. Regarding the distribution of CRLs, we implement a CRL pulling. We introduced a simple request-response mechanism.

For the performance evaluation and comparison we have used the wireless package of the J-SIM simulator. As we have already mentioned, we have implemented ADOPT, CPC-OCSP and CRLs. Through a practical, proof of concept, implementation of various validation schemes in a MANET-based environment, we demonstrated how ADOPT, a validation protocol that uses cached OCSP responses, can perform better than other relevant solutions. ADOPT, originally designed for MANETs can be applied to other forms of wireless, mobile and mesh networking in order to provide fresh CSI in an efficient in terms of overhead and security manner.

3 Conclusions

Apparently, OCSP-based schemes seem to suit better to restricted environment when compared to CRLs, SCVP and other related standards. Development of a completely new certificate validation solution, designed for use in wireless and mobile computing, would seem out of place, at the time when numerous established solutions exist. However, existing specifications provide mostly technical details, without giving sufficient information regarding certificate validation service discovery, caching capabilities and policies, time parameters and more. OCSP is an appealing solution but clearly, alterations should be made for it to fit in a highly dynamic and at the same time restricted environment. ADOPT takes on OCSP in order to provide a complete certificate validation solution which can be practically deployed in any network that supports broadcasting, offering interoperability between various platforms.

Finally, we discussed and evaluated how ADOPT, a certificate validation scheme that is based on a distributed version of OCSP, can be adapted and used for authentication and authorization purposes in VANETs. ADOPT, was originally

designed to work for MANETs, and disseminates certificate status information when requested, and periodically, such as CRL, d-CRL and over-issued CRLs. It takes into account network characteristics of VANETs, and prevents flooding of extended revocation lists, conserves the scarce bandwidth and avoids energy consumption that takes place during complex manipulations of revocation lists. It ties to overcome situations where Certificate Authorities cannot directly be communicated to provide status of certificates, due to network topology changes. It materializes efficient OCSP caching policies and short-length message bodies. We demonstrated using a practical implementation how the proposed scheme enhances the performance metrics, such as the delay in the location of the certificate's status, the freshness of status information, and the storage overheads, when compared to other CSI approaches, such as CRLs, SCVP and CPC-OCSP.

References

1. G. F. Marias, K. Papapanagiotou, V. Tsetsos, O. Sekkas and P. Georgiadis, "Integrating a Trust Framework with a Distributed Certificate Validation Scheme for MANETs", EURASIP Journal on Wireless Communications and Networking, Special Issue on Wireless Network Security. Vol. 2006.
2. K. Papapanagiotou, G. F. Marias, P. Georgiadis, "Revising Certificate Validation Standards for Mobile and Wireless Communications", Elsevier Computer Standards and Interfaces, Special Issue on Information and Communications Security, Privacy and Trust: Standards and Regulations. Vol. 2008 (under review).
3. K. Papapanagiotou, G. F. Marias and P. Georgiadis, "A Certificate Validation Protocol for VANETs", In Proceedings 2nd IEEE Workshop on Automotive Networking and Applications (AutoNet 2007), co-located with IEEE GLOBECOM 2007, Washington DC, USA, December 2007.
4. E. Kellinis and K. Papapanagiotou, "Using Steganography to Improve Hash Functions' Collision Resistance", In Proceedings SECRYPT 2007 International Conference on Security and Cryptography, Barcelona, July 2007.
5. K. Papapanagiotou, G. F. Marias, P. Georgiadis and S. Gritzalis, "Performance Evaluation of a Distributed OCSP Protocol Over MANETs," IEEE Consumer Communications and Networking Conference 2006 (IEEE CCNC06), Las Vegas, January 2006.
6. K. Papapanagiotou, E. Kellinis, G. F. Marias and P. Georgiadis, "Alternatives for Multimedia Messaging System Steganography," IEEE Computational Intelligence and Security (CIS), Xian, China, December 2005.
7. G. F. Marias, K. Papapanagiotou, and P. Georgiadis. Caching Alternatives for a MANET-Oriented OCSP Scheme, 1st IEEE/CREATE-NET Workshop on Security and QoS in Communication networks, Athens, Sept. 2005. K. Papapanagiotou, K. Markantonakis, Q. Zhang, W. G. Sirett and K. Mayes. On the performance of certificate revocation protocols based on a Java Card certificate client implementation. In Proceedings - 20th IFIP International Information Security Conference (Sec 2005) - Small Systems Security and Smart cards. Chiba, Japan, May 2005.
8. G. F. Marias, K. Papapanagiotou, and P. Georgiadis. ADOPT. A Distributed OCSP for Trust Establishment in MANETs, 11th European Wireless Conference 2005, Nicosia, Cyprus, April 2005.
9. G. F. Marias, K. Papapanagiotou, and P. Georgiadis. A Distributed OCSP Framework For Ad-Hoc Networks, International Conference in Applied Computing 2005, Algarve, Portugal, February 2005.