ADVERTIMENT. La consulta d'aquesta tesi queda condicionada a l'acceptació de les següents condicions d'ús: La difusió d'aquesta tesi per mitjà del servei TDX (www.tesisenxarxa.net) ha estat autoritzada pels titulars dels drets de propietat intel·lectual únicament per a usos privats emmarcats en activitats d'investigació i docència. No s'autoritza la seva reproducció amb finalitats de lucre ni la seva difusió i posada a disposició des d'un lloc aliè al servei TDX. No s'autoritza la presentació del seu contingut en una finestra o marc aliè a TDX (framing). Aquesta reserva de drets afecta tant al resum de presentació de la tesi com als seus continguts. En la utilització o cita de parts de la tesi és obligat indicar el nom de la persona autora.

ADVERTENCIA. La consulta de esta tesis queda condicionada a la aceptación de las siguientes condiciones de uso: La difusión de esta tesis por medio del servicio TDR (www.tesisenred.net) ha sido autorizada por los titulares de los derechos de propiedad intelectual únicamente para usos privados enmarcados en actividades de investigación y docencia. No se autoriza su reproducción con finalidades de lucro ni su difusión y puesta a disposición desde un sitio ajeno al servicio TDR. No se autoriza la presentación de su contenido en una ventana o marco ajeno a TDR (framing). Esta reserva de derechos afecta tanto al resumen de presentación de la tesis como a sus contenidos. En la utilización o cita de partes de la tesis es obligado indicar el nombre de la persona autora.

WARNING. On having consulted this thesis you're accepting the following use conditions: Spreading this thesis by the TDX (www.tesisenxarxa.net) service has been authorized by the titular of the intellectual property rights only for private uses placed in investigation and teaching activities. Reproduction with lucrative aims is not authorized neither its spreading and availability from a site foreign to the TDX service. Introducing its content in a window or frame foreign to the TDX service is not authorized (framing). This rights affect to the presentation summary of the thesis as well as to its contents. In the using or citation of parts of the thesis it's obliged to indicate the name of the author

UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Certificate Status Information Distribution and Validation in Vehicular Networks

by

Carlos Hernández Gañán

Advisor: José L. Muñoz Tapia
Co-Advisor: Oscar Esparza Martín

A thesis presented to the Department of
Telematics Engineering
in fulfillment of the requirement for the Degree of
Doctor of Philosophy
of the
Universitat Politècnica de Catalunya (UPC)

Barcelona, Spain, July 2013

Acknowledgements

Words fall short to express my gratitude and appreciation to my advisor José Luis Muñoz and Oscar Esparza, whose expertise, understanding, and patience, added considerably to my graduate experience. I appreciate his vast knowledge and skills in many areas, and his assistance in writing reports. Above all and the most needed, he provided me unflinching encouragement and support in various ways. His truly scientist intuition has made him as a constant oasis of ideas and passions in science, which exceptionally inspire and enhance my growth as a student, a researcher and a scientist want to be.

I would like also to thank and express my appreciation to the examining committee members. I appreciate their time and effort devoted for reading my Ph.D. thesis and providing me with their insightful comments and invaluable suggestions, thereby further improving the quality of my research work.

I am deeply indebted to Jonathan Loo. Without his guidance, support and good nature, I would never have been able to develop this thesis successfully. I benefited greatly from his ideas and insights. His involvement with his originality has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come.

Some debts are hard to put into words. My research colleagues Juan Caubet, Sergi Reñé, Jorge Mata, Juan José Alins all know why their names are here.

My last, but not least gratitude is for my parents, it is difficult to find words to express my gratitude and thanks to both of you.

I realize that not all people who contributed either directly or indirectly to my study are mentioned in this page. From the deepest of my heart, I would like to thank all of you...

Abstract

Vehicular ad hoc networks (VANETs) are emerging as functional technology for providing a wide range of applications to vehicles and passengers. Ensuring secure functioning is one of the prerequisites for deploying reliable VANETs. However, the open-medium nature of these networks and the high-speed mobility of a large number of vehicles harden the integration of primary security requirements such as authentication, message integrity, non-repudiation, and privacy.

Without security, all users would be potentially vulnerable to the misbehavior of the services provided by the VANET. Hence, it is necessary to evict compromised, defective, and illegitimate nodes. The basic solution envisioned to achieve these requirements is to use digital certificates linked to a user by a trusted third party. These certificates can then be used to sign information. Most of the existing solutions manage these certificates by means of a central Certification Authority (CA). According to IEEE 1609.2 standard, vehicular networks will rely on the public key infrastructure (PKI). In PKI, a CA issues an authentic digital certificate for each node in the network. Therefore, an efficient certificate management is crucial for the robust and reliable operation of any PKI. A critical part of any certificate-management scheme is the revocation of certificates. The distribution of certificate status information process, as well as the revocation process itself, is an open research problem for VANETs.

In this thesis, firstly we analyze the revocation process itself and develop an accurate and rigorous model for certificate revocation. One of the key findings of our analysis is that the certificate revocation process is statistically self-similar. As none of the currently common

formal models for revocation is able to capture the self-similar nature of real revocation data, we develop an autoregressive Fractional-integrated moving average (ARFIMA) model that recreates this pattern. Neglecting the self-similarity of the revocation process leads to inefficient revocation release strategies. With synthetic revocation traces, current revocation schemes can be improved by defining more accurate revocation data issuance policies. We show that traditional mechanisms that aim to scale could benefit from these traces to improve their updating strategies.

Secondly, we analyze how to deploy a certificate status checking service for mobile networks and we propose a new criterion based on a risk metric to evaluate cached status data. With this metric, the PKI is able to code information about the revocation process in the standard certificate revocation lists. Thus, users can evaluate a risk function in order to estimate whether a certificate has been revoked while there is no connection to a status checking server. Moreover, we also propose a systematic methodology to build a fuzzy system that assists users in the decision making process related to certificate status checking.

Thirdly, we propose two novel mechanisms for distributing and validating certificate status information (CSI) in VANET. This first mechanism is a collaborative certificate status checking mechanism based on the use based on an *extended-CRL*. The main advantage of this *extended-CRL* is that the road-side units and repository vehicles can build an efficient structure based on an authenticated hash tree to respond to status checking requests inside the VANET, saving time and bandwidth. The second mechanism aims to optimize the trade-off between the bandwidth necessary to download the CSI and the freshness of the CSI. This mechanism is based on the use of a hybrid delta-CRL scheme and Merkle hash trees, so that the risk of operating with unknown revoked certificates remains below a threshold during the validity interval of the base-CRL, and CAs have the ability to

manage this risk by setting the size of the delta-CRLs. For each of these mechanism, we conduct security analysis and performance evaluation to demonstrate the reliable security and efficiency of the proposed schemes.

Finally, we also analyze the impact of the revocation service in the certificate prices. We model the behavior of the oligopoly of risk-averse certificate providers that issue digital certificates to clients facing identical independent risks. We found the equilibrium in the Bertrand game. In this equilibrium, we proof that certificate providers that offer better revocation information are able to impose higher prices to their certificates without sacrificing market share in favor of the other oligarchs.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Today's transportation systems face serious challenges in terms of road safety, efficiency and environmental friendliness. With a huge improvement in technological innovations, Vehicular Communication (VC) emerges as a solution to palliate many issues of our modern day communication system in roads. This type of communication involves the use of short-range radios in each vehicle. This technology allows various vehicles to communicate with each other which is also known as (V2V) communication and with road side infrastructure (V2I) communication. Vehicular communication systems (VCS) are a direct response to the increasing demands of Intelligent Transportation Systems (ITS) services and the expectations of the automotive industry. In this sense, vehicular communication is designed for a wide range of applications related to safety, traffic management, and passenger comfort.

Safety applications are the main motivation for the development of these systems. They are conceived to spread accurate data quickly and reliably, in order to avoid accidents and life losses. In this sense, vehicles collaborate to avoid accidents, e.g., they disseminate emergency warning messages when a hazardous status is detected, such as slippery road conditions. In the same way, VCSs improve road safety by enabling traffic lights and signs to communicate with vehicles. In addition to these safety applications, VCSs are also employed in a variety of ITS traffic management applications. Road traffic management applications focus on optimizing traffic flow in order to avoid traffic congestion, to reduce travel time, and to use the transportation infrastructure effectively. A third type

of applications relates to the comfort and well-being of passengers, named infotainment applications. Infotainment applications provide additional information or entertainment to the passengers and/or driver, e.g. multimedia services, radio channels, Internet connection or advertises from some local merchants or gas stations.

Vehicular networks have attracted the attention of both academic and industrial communities, which is reflected in the interest of governments and standardization organizations. For example, European car manufacturers have instituted the Car-to-Car Communication Consortium (C2C-CC) [2] to improve road safety and efficiency, and the U.S FCC (Federal Communication Commission) has approved a 75 MHz spectrum for vehicular networks [3]. The Institute of Electrical and Electronics Engineers (IEEE) also supports vehicular communication with the IEEE 1609 family of standards for wireless access in vehicular environments (WAVE) [4]. Previous works present approaches that employ various technologies for the implementation of VCS. In this way, several car manufacturers support their vehicles through Internet access via cellular networks. However, using cellular networks is not the best way to build a VCS in terms of cost and latency. In many proposals, standard IEEE 802.11 is deployed for a VCS. However, this protocol has a limited radio range and needs numerous base stations to maintain the vehicles connected to the infrastructure. Using Vehicular Ad Hoc Network (VANET) with On-Board Units (OBUs) and Roadside Units (RSUs) appears to be the more effective method, but it also entails significant challenges. VANETs also enable multi-hop routing through vehicles to reach the infrastructure. Nevertheless, without securing these networks, damage to property and life can be done at a greater extent.

Simple and effective security mechanisms are the major problem of deploying VANET in public. Without security, a VANET is wide open to a number of attacks such as propagation of false warning messages as well as suppression of actual warning messages, thereby causing accidents. This makes security a factor of major concern in building such networks.

1.1 Research Motivation

VANETs deployment will not occur without assuring secure communications. As a special case of mobile ad hoc networks (MANETs), VANETs inherit all of MANETs security concerns while introducing additional security challenges specific to their characteristics. In VANETs, attackers could forge, inject, replay and drop messages in order to violate user privacy, information integrity, authenticity, and system performance.

In order to secure a VANET, the following security requirements should be met [5]:

- *Authentication*: Entity authentication is required to ensure that the communicating entities are legitimate. In addition, data authentication is also a concern to ensure that the contents of the received data is neither altered nor replayed.

- *Non-repudiation*: Non-repudiation is necessary to prevent legitimate users from denying the transmission or contents of their messages.

- *Privacy*: Preserving users' privacy is necessary to prevent the disclosure of their location information and real identities.

- *Access control*: Access control is required to delimitate the operations that any entity in the network is allowed to perform. Moreover, any misbehaving entity should be removed from the network to protect other legitimate entities. In addition, any action taken by those misbehaving entities should be repealed.

- *Availability*: Users may be frustrated if VANET services become temporarily unavailable due to attacks such as DoS attacks.

Without security, all users would be potentially vulnerable to the misbehavior of the services provided by the VANET. Hence, it is necessary to evict compromised, defective, and illegitimate nodes. The basic solution envisioned to achieve these requirements is to use digital certificates linked to a user by a trusted third party. These certificates can then be used to sign information. Most of the

existing solutions manage these certificates by means of a central Certification Authority (CA) [6, 7]. According to IEEE 1609.2 standard [8], vehicular networks will rely on the public key infrastructure (PKI). In PKI, a CA issues an authentic digital certificate for each node in the network. Therefore, an efficient certificate management is crucial for the robust and reliable operation of any PKI. A critical part of any certificate-management scheme is the revocation of certificates.

To revoke a vehicle in PKI, a certificate revocation list (CRL) has to be issued by the trusted authority (i.e., centralized revocation) and broadcast by the infrastructure RSUs. The centralized certificate status checking process in the classical PKI may be impractical in the large scale VANETs due to the following reasons: (1) Each CA encounters a large number of CRL requests which can render the CA a bottle-neck; (2) The CRL downloading process is long relative to the short V2I communication duration between the immobile RSUs and the highly mobile OBUs during which the new CRL should be delivered to the requesting OBU. This long delay is due to the fact that a request submitted by an OBU to an RSU must be forwarded to the CA, and CA has to send the new CRL to that RSU which in turn forwards the new CRL to the requesting OBU. Accordingly, the classical PKI should be optimized to satisfy the revocation service requirement in vehicular communication scenarios. To provide a practical revocation service for VANETs, it is required for each OBU to efficiently check the status of any certificate in a timely manner.

Additionally, while wired networks can guarantee on demand connectivity between the CA and principals in the wired network for obtaining the current certificate status information, VANETS cannot guarantee such on demand contact at all times due to the sporadic connectivity. Such on demand connectivity with the centralized entity is essential for recipients to have confidence on the security infrastructure. Hence, the risk on trusting outdated certificate status information while being disconnected from the infrastructure needs to be quantified.

1.2 Objective of the Thesis

This thesis aims to mitigate the issues of conveying certificate status information over vehicular networks. Therefore, the objectives of this thesis are as follows:

1. Modeling the revocation process to perform analytic and empirical evaluations. This modeling should allow users and authorities to predict when a revocation is prone to occur. Moreover, the resulting model should also serve as evaluation tool to generate synthetic revocations.

2. Evaluating the performance of current certificate status mechanisms in vehicular networks. This involves analyzing the impacts of sporadic connectivity with the security infrastructure on the security performance of the proposed security mechanisms.

3. Proposing a metric to quantify the risk of operating while being disconnected from the infrastructure. This metric should take into account all the information about the revocation process available to the certification authority.

4. Designing a new certificate status validation mechanism for vehicular networks. This mechanism should take into account the knowledge derived from the revocation process modeling, and use the criterion to measure the risk of operating while disconnected from the infrastructure.

Chapter 2

Results

This section summarizes the main contributions of this thesis aligned with the objectives detailed in the previous section. Throughout the research process that involves the development of the thesis, several results have been obtained. These results have been validated by the international scientific community through the assessment of papers published in high-ranked journals and international conferences. Each contribution is briefly described in the following sections and appended at the end of this document.

2.1 Analysis and modeling of the revocation process

Most of the effort on analyzing certificate revocation has been mainly put on studying the trade-offs that can be achieved when dealing with different revocation mechanisms [9, 10, 11, 12]. These studies aim to compare the performance of different revocation mechanisms in different scenarios. Recently however, there have appeared some studies like [13, 14, 15] that can be considered a first step towards understanding the revocation process itself. These studies have mainly analyzed the probability distribution of certificate revocation requests. However, these later studies do not capture the time evolution of the revocation process or provide a means to efficiently forecast revocation events.

A revocation method is selected by an organization based on the cost, infrastructure, and volumes of transactions that are expected. To gauge these costs, different revocation mechanisms are tested under the assumption that the revocation events follow a specific probability distribution. Most theoretical frameworks and simulation studies for performance evaluation assume that the temporal distribution of queries follows a Poisson distribution and using this, organizations can estimate the infrastructure needed to deploy the PKI and the associated costs. However, in this thesis, we have demonstrated that revocation data is statistically *self-similar*, that none of the commonly used revocation models is able to capture this fractal behavior, and that such behavior has serious implications for the design, control, and analysis of revocation mechanisms such as CRLs.

We started by analyzing the validity of Poisson-like process assumption. We used publicly available CRLs from different certification authorities (containing more than 300,000 revoked certificates over a period of three years (see Table 2.1)). Our analysis demonstrated that the Poisson distribution fails to capture the statistical properties of the actual revocation process. We also saw that the Poisson distribution grossly under-estimates the bandwidth utilization of the revocation mechanism. At first glance, this might look like an obvious result, since after all as a memoryless process, Poisson distribution cannot be expected to model periodic trends like daily, weekly and monthly cycles in revocation rates. We showed however that the modeling inability transcends simple cycles. In particular, we showed self-similarity has a severe detrimental impact on the revocation service performance.

Issuer Name	Number of Revoked Certificates	Last Update	Next Update
GoDaddy	932,900	2012/02/01	2012/02/03
VeriSign	5,346	2012/02/02	2012/02/16
Comodo	2,727	2012/02/03	2012/02/06
GlobalSign	7,591	2012/02/02	2012/03/03
Thawte	8,061	2012/02/01	2012/02/16

Table 2.1: Description of the collected CRLs.

Results of our analysis, including burstiness at all scales, strongly indicate self-similar nature of revocation events. In Figure 2.1 we can observe different evident trends; (i) Burstiness in all time scales: the burstiness of the revocation process does not disappear when changing the time scales. (ii) Lack of natural length of bursts: The figure shows burstiness ranging from days to months. Note that the full duration of the figure with the largest time slot is 1,000 days, and some of the bursts have many hours of duration.

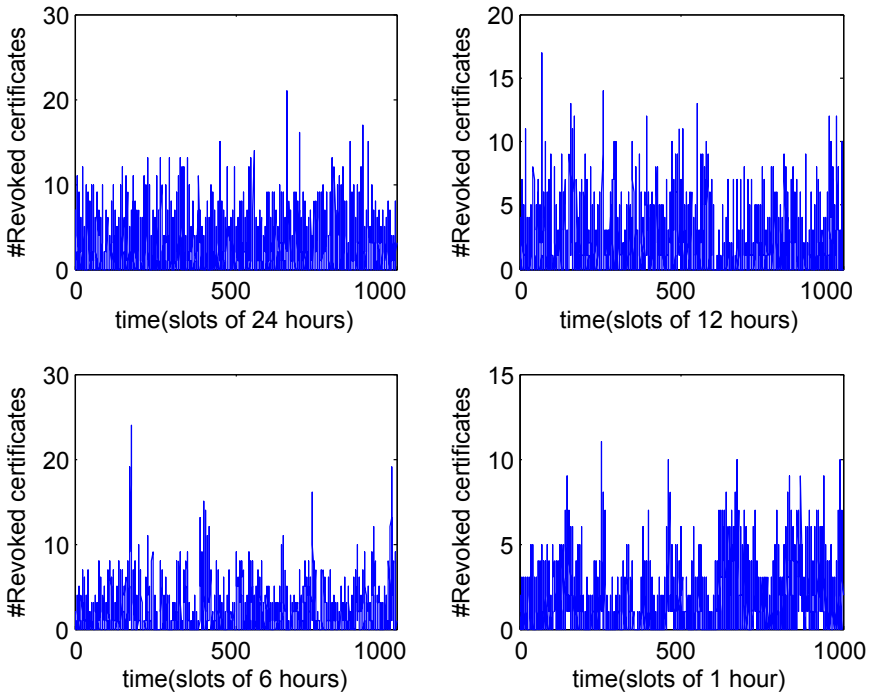


Figure 2.1: Revocation Bursts over Four Orders of Magnitude.

We confirmed this by analyzing the autocorrelation of the revocation process and estimating the Hurst parameter for the observed distribution and showing that the estimates validate self-similar nature of the revocation lists. First of all, we started analyzing the autocorrelation of the revocation data. Recall that in a self-similar process autocorrelations decay hyperbolically rather than exponentially fast, implying a nonsummable autocorrelation function $\sum_k r(k) = \infty$ (long-range dependence or LRD). For the frame data, the empirical autocorrelation functions $r(k)$ are shown in Fig. 2.2, with lag k ranging from 0 to 100. Notice

that $r(k)$ decreases slower than exponentially no matter the CA. The curve does decay toward zero, but it does so extremely slowly. The very slowly decaying autocorrelations are indicative of LRD.

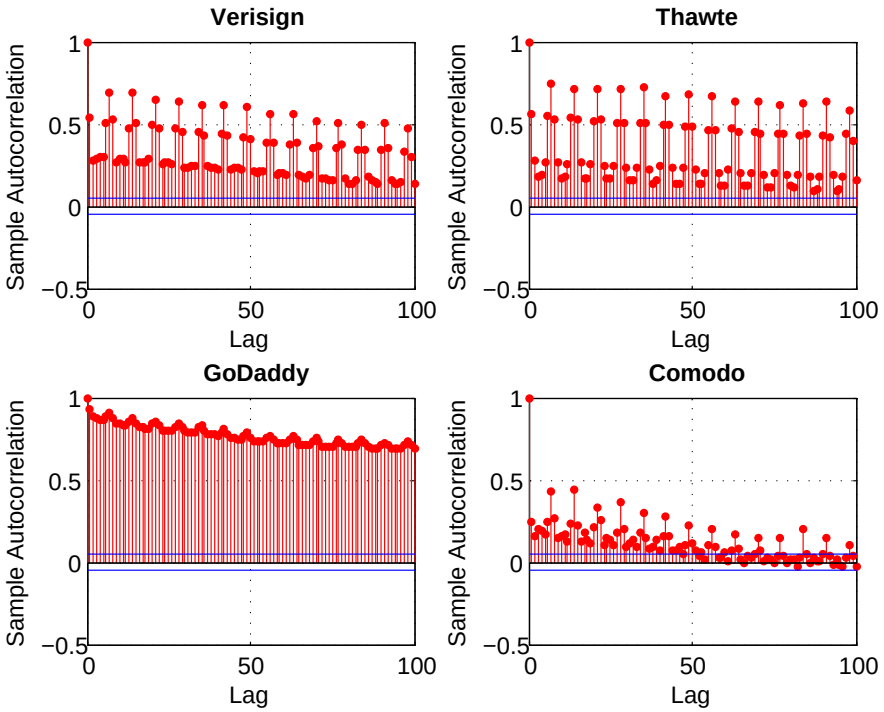


Figure 2.2: Autocorrelation function of the revocation process per CA.

Then, we used five different methods for assessing self-similarity: the variance-time plot, the rescaled range (or R/S) plot, the periodogram plot, the Detrended Fluctuation Analysis (DFA) plot and the Whittle estimator. We concentrated on individual months from our revocation time series, so as to provide as nearly a stationary dataset as possible. To provide an example of these approaches, analysis of a single month from GoDaddy revocation data is shown in Figure 2.3. The figure shows plots for the four graphical methods: variance-time (upper left), rescaled range (upper right), periodogram (lower left) and DFA (lower right). The variance-time plot is linear and shows a slope that is distinctly different from -1 (which is shown for comparison); the slope is estimated using regression as -0.077, yielding an estimate for H of 0.96. The R/S plot shows an asymptotic slope that is different from 0.5 and from 1.0 (shown for comparison); it is estimated

using regression as 0.95, which is also the corresponding estimate of H. The periodogram plot shows a slope of -0.14 (the regression line is shown), yielding an estimate of H as 0.83. Finally, the Whittle estimator for this revocation data (not a graphical method) yields an estimated Hurst value of 0,923 with a 95% confidence interval of $(0.87, 0.95)$.

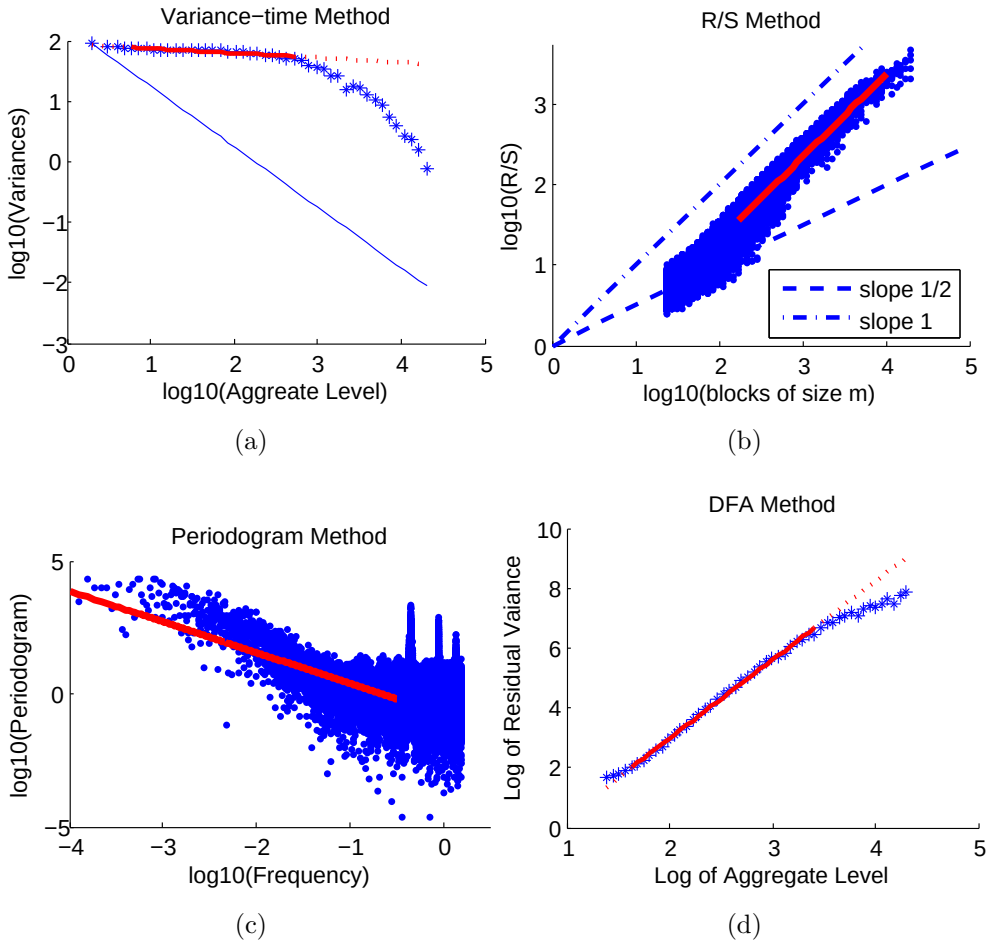


Figure 2.3: Graphical methods for checking for self-similarity of the revocation process from GoDaddy (a) variance-time plot, (b) pox plot of R/S, (c) periodogram plot, and (d) DFA plot.

Beyond invalidating Poisson-like distributions, this proof of self-similarity has important implications on CA utilization, throughput, and certificate stratus checking time. Intuitively, as the revocation process is bursty (non-uniformly

distributed) the CA will be partially idle during low burst periods and vice versa. Thus, the revocation lists will grow non-uniformly, and current updating policies will result bandwidth inefficient.

After proving the selfsimilar nature of the revocation process, we went a step further by developing an accurate and rigorous model for certificate revocation process. The proposed model is based on an autoregressive fractionally integrated moving average (ARFIMA) process [16], which provides an accurate and parsimonious model for revocation.

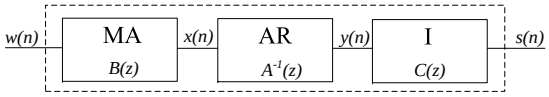


Figure 2.4: Components of an ARFIMA process.

Figure 2.4 shows a scheme of the ARFIMA model, where the components of each bloc are:

$$\begin{aligned} A(z) &= 1 - 0.6467z^{-1} + 0.02693z^{-2} + 0.09085z^{-3} + 0.09753z^{-4} + 0.1218z^{-5} + 0.1991z^{-6} \\ &\quad - 0.804z^{-7} + 0.6906z^{-8} + 0.03223z^{-9} - 0.04807z^{-10} - 0.007471z^{-11} - 0.0759z^{-12} \\ &\quad - 0.08934z^{-13} - 0.07605z^{-14} - 0.006487z^{-15} - 0.02565z^{-16} - 0.01994z^{-17} - 0.04003z^{-18} \\ &\quad - 0.05007z^{-19} - 0.01331z^{-20} - 0.07361z^{-21} - 0.001947z^{-22} - 0.02836z^{-23} - 0.01824z^{-24} \\ &\quad - 0.03693z^{-25} + 0.007019z^{-26} - 0.07691z^{-27} - 0.01872z^{-28} - 0.03821z^{-29}, \qquad (2.1) \\ B(z) &= 1 - 0.6454z^{-1} + 0.005554z^{-2} + 0.1113z^{-3} + 0.1317z^{-4} + 0.1032z^{-5} + 0.2802z^{-6} \\ &\quad - 0.6652z^{-7} + 0.6688z^{-8}, \qquad (2.2) \\ C(z) &= (1 - z^{-1})^{-0.3}. \qquad (2.3) \end{aligned}$$

Once we obtained the model, we described how to use it to build a synthetic revocation generator that can be used in simulations of resource assessment. To be able to construct the revocation trace generator, we needed to concatenate a zero-memory non-linear function (ZMNL) to the ARFIMA model Figure 2.5 shows the block diagram of the synthetic revocation generator, where the ZMNL function is placed at the output of the ARFIMA filter. The values of the white noise sequence $w(n)$ at the input of the ARFIMA filter are chosen such that $Var(w(n)) = 1$ and $E[w(n)] = 0$. In turn, the output of the ARFIMA filter $s(n)$ becomes the input

of the ZMNL function. In this way, the ARFIMA model transforms the $N(0,1)$ sequence in a colored $N(0,\sigma_s^2)$ sequence. Then, the ZMNL function transforms the colored $N(0,\sigma_s^2)$ sequence in an $Exponential(\mu_r)$ sequence, where μ_r is the measured average of daily revoked certificates.

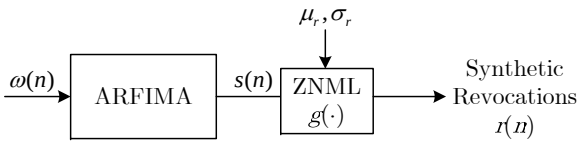


Figure 2.5: Synthetic Revocation trace generator.

Hence our model produces synthetic revocation traces that are indistinguishable for practical purposes from those corresponding to actual revocations.

2.2 PKI deployment in vehicular adhoc networks

Our previous results showed that when deploying revocation mechanisms the characteristics of the revocation process have to be taken into account. In the case of the vehicular networks, the IEEE 1609.2 standard [8] states that they will rely on the use of CRLs. In particular, OBUs must obtain the Certificate Status Information (CSI) from the revocation system. In the literature there are several mechanisms to distribute CSI in environments prone to disruptions [10, 17, 18, 19] though none of them takes into account the revocation process characteristics in its design. They are essentially based on retrieving the CSI from the infrastructure during connectivity intervals and using some caching strategy when the connection to the infrastructure is not possible. Then, OBUs may use their cached copy of the CSI (previously downloaded during a connectivity interval) or may try to discover more recent CSI among their neighbors. Once a copy of the CSI is obtained, OBUs have to face the problem of evaluating the freshness of this copy. Depending on the freshness of the CSI, the risk of trusting this information as comprehensive will vary. OBUs should be able to quantify

this risk to make an informed decision whether to operate or not with a specific certificate.

CRLs are expected to be quite large because the network scale of VANETs is expected to be very large and because to protect the privacy of users each vehicle is going to have many temporary certificates (or pseudonyms). Hence, the distribution of CRLs is prone to long delays. Moreover, during the early deployment of VANETs, RSUs may not be uniformly distributed in the network. Therefore, the way of distributing CRLs must be designed to ensure that revocation is can be correctly deployed in those delay-tolerant environments. There have been proposed several ways to improve the distribution of CRLs (e.g. [7, 19]). These proposals intended to make more efficient the distribution of the CRLs, by for example, reducing its size or using V2V communications. However, none of these proposals deals with the problem of the lack of information about certificates that are revoked during the validity interval of a CRL. In this thesis, we presented a metric that quantifies the risk that the recipients are facing when accepting messages signed with certificates that are not present in the CRLs at the OBU.

Using group theory and a probabilistic analysis, we calculated the probability of considering a certificate as a valid one when the real status known by the CA is revoked at time t as (see details in [20]):

$$\rho(t) = Prob(Cert \in \mathcal{U}) = \frac{p(t - t_0)}{(1 - p)T_c + p(t - t_0)}, \tag{2.4}$$

where T_c is the mean certificate lifetime, p is the percentage of revoked certificates and $\mathcal{U}$ is the set of revoked certificates that were not included in the previous CRL.

Figure 2.6 shows the theoretical evolution of $\rho(t)$ during three consecutive CRL updates. As expected, the probability is zero at instants of CRL update as there are no unknown revoked certificates. On the contrary, this probability is maximum just before publicizing a new CRL, as the number of unknown revoked certificates is maximum at this point. Note that this maximum (as well as the slope of the probability function) varies depending of the percentage of revoked

certificates (p_i). Thus, when this percentage is higher (note that $p_2 > p_3 > p_1$) the probability increases more rapidly.

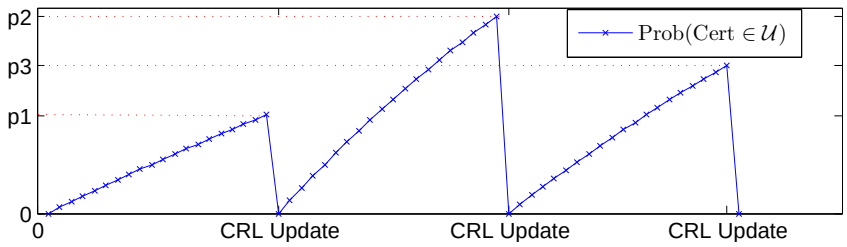


Figure 2.6: Time evolution of the probability of considering an unknown revoked certificate as valid.

Once we obtained the probability of operating with an unknown revoked certificates, we developed a new risk analysis method to identify and assess this risk. In addition, we must assure that risk information produced is processed and reliably applied to decision making. Previous works in the literature [21, 22, 23] acknowledged the existence of an operational risk when using a revocation mechanism such as CRLs. However, these works neither quantified this risk nor provided a means to deal with it. We modeled and characterized a risk-based decision making system based on fuzzy logic. To that end, taking into account the information that users can obtain from the CRLs, we design a fuzzy inference system that gives as output the risk of operating with a particular CRL. Using the proposed model, users get an idea of how risky is to operate with their current CRL and are able to make risk-based decisions.

Risk analysis by the trusting user in its potential interaction with a probable illegitimate user was done by:

1. Determining the possibility of operating with users that have their certificate revoked using $\rho(t)$;

2. Determining the possible consequences of operating with an illegitimate user, using the certificate revocation causes (see Table 2.2).

Then, we defined a fuzzy-risk based decision making system where the inputs of the inference system were:

Numerical Code	Revocation Code	w_i	Description
(1)	`keyCompromise`	9	Private key has been compromised.
(2)	`cACompromise`	10	Certificate authority has been compromised.
(3)	`affiliationChanged`	1	Subject's name or other information has changed.
(4)	`superseded`	0	Certificate has been superseded.
(5)	`cessationOfOperation`	1	Certificate is no longer needed.
(6)	`certificateHold`	3	Certificate has been put on hold.
(7)	`removeFromCRL`	0	Certificate was previously on hold and should be removed from the CRL.
(8)	`privilegeWithdrawn`	5	Privileges granted to the subject of the certificate have been withdrawn.
(9)	`aACompromise`	10	Attribute authority has been compromised.

Table 2.2: Revocation codes, weight values w_i and description.

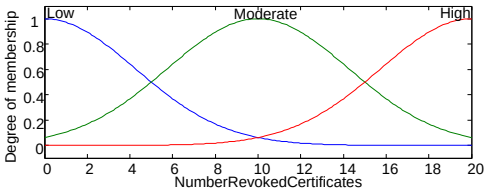
1. **Number of revoked certificates** ($NumRev$): as users have cached CRLs which include the list of revoked certificates and their revoked date, users can know the number of revoked certificates per day;

2. **Revocation categories** ($RevCat$): CRLs can also include the revocation cause of each certificate;

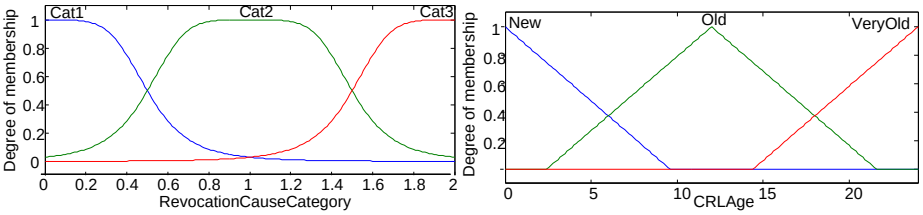
3. **Age of the CRL** (CRL_{age}): using also the information contained in the CRL; users can calculate the time elapsed since the issuance of the CRL.

For each of these inputs we defined a *membership* function (see Fig. 2.7).

Finally, a case study on risk analysis of a CRL issued by an actual CA was used to show the validity of the proposed model. The results of the risk assessment in the case study were represented as risk score, located in a defined range, and risk category with linguistic words, which indicates that by using the proposed methodology the risk associated with CRLs can be assessed effectively

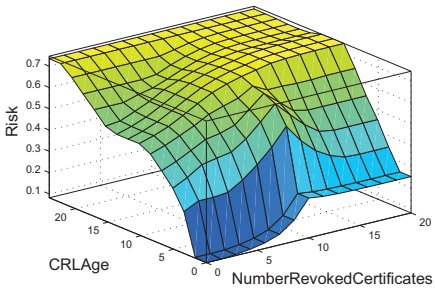


(a) Number of Revoked Certificates

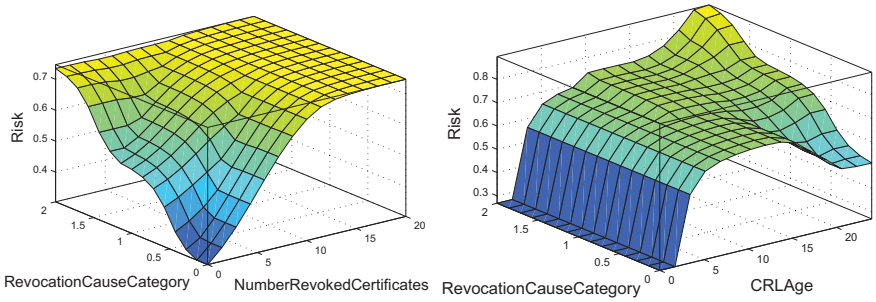


(b) Revocation Cause Categories (c) CRL Age

Figure 2.7: Membership functions.



(a)



(b) (c)

Figure 2.8: Risk Indicator as a combination of a) the CRL age and the number of revoked certificates, b) the revocation cause categories and the number of revoked certificates, c) the CRL age and the revocation cause categories.

and efficiently. These results showed that although this CA is issuing CRLs with a frequency of 1 day, there is still some inherent risk that our model achieves to measure (see Fig. 2.8). Based on this metric, any user that operates using certificates from this CA could make informed decisions.

2.3 Certificate Status Checking mechanism for VANETs

At this point of the thesis, we were aware that current revocation mechanisms will exhibit some issues when directly applied to a vehicular network, and they need to be improved in terms of efficiency and CSI freshness. First of all, we analyzed the drawbacks of applying the IEEE 1609.2 standard proposal which suggests the use of CRLs.

As mentioned before, for a CA to invalidate a vehicle's certificates, the CA includes the certificate serial number in the CRL. The CA then distributes the CRL so that vehicles can identify and distrust the newly revoked vehicle. The distribution should spread quickly to every vehicle in the system. However, the distribution itself poses a great challenge due to the size of the CRL. As a CRL is a list containing the serial numbers of all certificates issued by a given certification authority that have been revoked and have not yet expired, its distribution causes network overhead. Moreover, the CRL size increases dramatically even if only a small portion of the OBUs in the VANET is revoked. To have an idea of how big the CRL size can be, consider the case where 1% of the total number of the OBUs in the United States is revoked. Recall that in a VANET, each vehicle owes not only an identity certificate, but also several pseudonyms. The number of pseudonyms may vary depending on the degree of privacy and anonymity that it must be guaranteed. According to [24], OBUs must store enough pseudonyms to change pseudonyms about every minute while driving. This equates to about 43,800 pseudonyms per year for an average of two hours of driving per day. In the US, 255,917,664 "highway" registered vehicles were counted in 2008, of which 137,079,843 passenger cars [25]. In this case, the CRL would contain around 100 billion revoked certificates. Assuming that certificates can be identified by a 16

byte fingerprint (the size of one AES block), the CRL size is around 1,7 TB. Only the amount of memory necessary to storage this CRL makes it impossible its deployment.

The CRL size can be reduced by using regional CAs. However, there appears a trade-off between the size of the CA region and size of the CRL, as well as the management complexity of the entire PKI system for VANETs. The least complicated region to manage would be a single large area, such as the entire United States, with a single CA responsible for every certificate and pseudonym. However, this gives place to CRLs of several terabytes. Hence, it is necessary to divide the CRL information according to regional areas. In this sense, if we divide the entire United States by cities (i.e. ∼10,016 cities), the CRL size is reduced to around 170 Mbytes. Using the 802.11a protocol to communicate with RSUs in range, vehicles could have between 10-30 Mbps depending on the vehicle's speed and the road congestion [26]. Thus, in the best case (under non-congested conditions) a vehicle will need more than 45 seconds to download the whole CRL. In scenarios where vehicles are not able to keep a permanent link with the infrastructure for this amount of time, techniques such as Bloom filter or Digital Fountain Codes could be used to download the CRL. Therefore, though the problem of having a huge CRL is mitigated by the use of such techniques, the restraints imposed by the distribution affect the freshness of the revocation data.

A direct consequence of this significant time to download a CRL is that a new CRL cannot be issued very often, so its validity period has to be shortened. This validity period directly determines how often a vehicle has to update the revocation data. Therefore, the validity period of the CRL is critical to the bandwidth consumption. Moreover, it appears another trade-off between the freshness of revocation data and the bandwidth consumed by downloading CRLs. Large validity periods will decrease the network overhead at expenses of having outdated revocation data. Small validity periods will increase the network overhead but users will have fresh information about revoked certificates. As CRLs cannot be issued every time there is a new revoked certificate, vehicles will be operating with revocation data that are not comprehensive. In this thesis we developed a revocation mechanism to improve the performance of the revocation process in a

vehicular network by taking advantage of authenticated data structures and V2V communications.

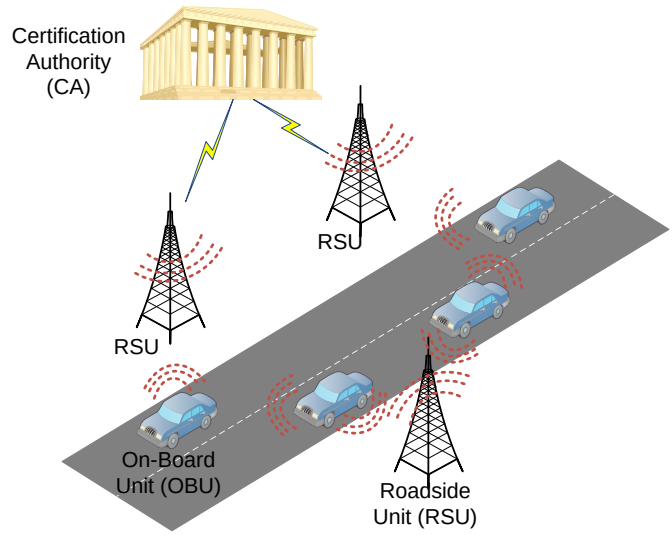


Figure 2.9: System Architecture.

Our proposal was called COACH (COllaborative certificate stAtus CHecking). COACH is an application-layer mechanism for distributing revocation data. The system architecture is an adaptation of a PKI system to the vehicular environment (see Fig. 2.9). The main idea behind COACH is to embed some little extra information into the CRL such that allows us to create an efficient and secure request/response protocol. For those nodes that just need to obtain status data of some certificates, our protocol avoids downloading a complete CRL. Specifically, we proposed a way of efficiently embedding a Merkle hash tree (MHT) [27] within the structure of the standard CRL to generate a so-called *extended-CRL*.

To create the *extended-CRL*, we used an extension, which is a standard way of adding extra information to the CRL. Our extension contains all the necessary information to allow any vehicle or VANET infrastructure element that possesses the *extended-CRL* to build the COACH tree, i.e., a hash tree with the CSI of the CRL. Using this COACH tree, any entity possessing the *extended-CRL* can act as repository and efficiently answer to certificate status checking requests of other vehicles or VANET elements (see Fig. 2.10). COACH responses are short since in general, their size is less than 1 Kbyte (see Table 2.3, where T_{hash}

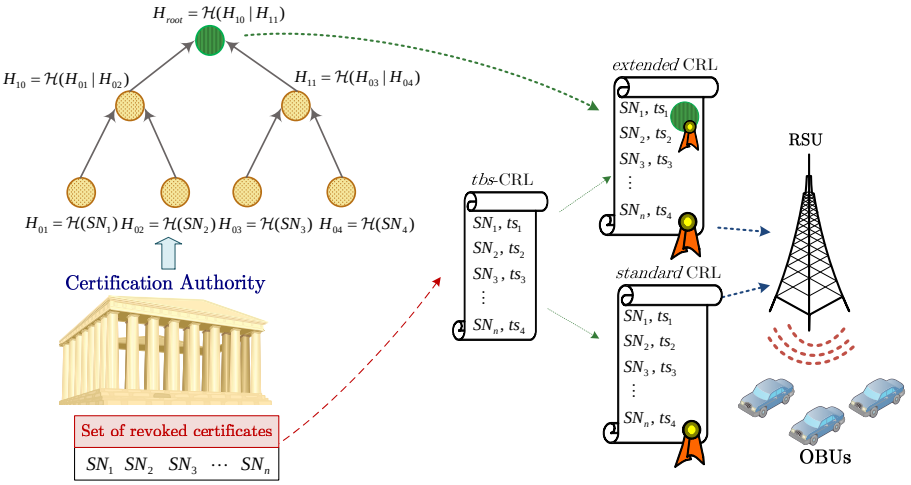


Figure 2.10: COACH bootstrapping.

and T_{mul} denote the time required to perform a pairing operation and a point multiplication, respectively). This allows a COACH response to perfectly fit within a single UDP message. We also proposed an enhancement of our basic mechanism called EvCOACH (Evergreen-COACH) to improve the performance of COACH in scenarios with relatively few revocations per CRL validity period.

Mechanism	Request size	Response size	Verification delay	Signing delay
CRL	73 bytes	145 Mbytes	$4T_{mul}$	T_{mul}
COACH	73 bytes	710 bytes	$k(T_{hash}(log_2 N + 1) + 4T_{mul})$	T_{mul}
EvCOACH	73 bytes	725 bytes	$k(T_{hash}(log_2 N + i + 2) + 4T_{mul})$	T_{mul}
ADOPT	66 bytes	586 bytes	$k(4T_{mul})$	$k(T_{mul})$

Table 2.3: COACH vs other certificate validation mechanisms

Not that ADOPT [10] (Ad-hoc Distributed OCSP for Trust) provides a revocation service based on the Online Status Checking Protocol (OCSP)[**?**] in a decentralized manner. ADOPT uses cached OCSP responses that are distributed and stored on intermediate nodes in the VANET. Thus, ADOPT's query cost is the lowest but not far from COACH. Moreover, we also showed by simulation,

that COACH makes the distribution of CSI more efficient than distributing complete CRLs (even though they are compressed), reducing the data that have to be transmitted over the VANET.

Vehicle Speed	***Delay***								
	COACH			**CRL**			**ADOPT**		
	Tx	*Comp.*	*RTT*	*Tx*	*Comp.*	*RTT*	*Tx*	*Comp.*	*RTT*
20 m/s	75 ms	2,401 ms	78 ms	3,521 h	2,400 ms	3,521 h	72 ms	3,612 ms	101 ms
30 m/s	149 ms	2,401 ms	157 ms	8,213 h	2,400 ms	8,214 h	122 ms	3,600 ms	312ms
40 m/s	173 ms	2,401 ms	187 ms	9,811 h	2,400 ms	9,813 h	152 ms	3,600 ms	421ms

Table 2.4: Delays when querying for CSI.

Table 2.4 shows the mean delays incurred when querying for the status of a given certificate. With *transmission delay* we denote the time to send the CSI query and the corresponding response. If we compare the transmission delay of the different revocation mechanisms, we can observe that ADOPT is the fastest but not so far from COACH. On the other hand, by *computational delay* we denote the time required to compose and validate a CSI response. In this case, ADOPT has the worst computational delay because each CSI response has to be signed by the CA. CRL computational delay is minimal as the CRL is only signed once and to searching the serial number of a certificate in the list has a computational cost of $O(log_2 N)$. COACH only requires one CA signature but a $\mathcal{P}$ath has to be computed each time a CSI response is required, so the computational cost is similar to the CRL. Finally, we define Round-Trip Time (RTT) as the time that takes since a vehicle requests for CSI until the status of a given certificate is validated. Therefore, the RTT is affected by the transmission, computational and propagation delays. ADOPT has the worst RTT due to the multihop transmission of the cached CSI, while CRL and COACH download the CSI directly from the repository in range. In any case, the vehicles' speed affects transmission and RTT delays in all three revocation mechanisms. We must stress that a node possessing an *extended-CRL* can act as COACH repository but that a COACH repository is not a TTP. In other words, COACH is cryptographically offline, which means that

no online trusted entity (like a CA) is needed for authenticating the responses produced by COACH repositories.

Though COACH improves the efficiency of the revocation mechanism, it does not enhance the freshness of the revocation data. To that end, we designed BECSI, a Bandwidth Efficient Certificate Status Information distribution mechanism for VANETs. BECSI improves the distribution of CSI by transmitting the revocation information that is unknown to a particular user during the validity period of the cached CSI. The main idea behind BECSI is to allow vehicles requesting for new CSI during the validity period of the current CRL. Thus, revocations that occur during the validity period of the CRL will not be unknown to the vehicles during this whole validity period, reducing the risk of operating with an unknown revoked certificate. We addressed the CRL distribution problem by exploiting the combination of three well-known mechanisms: (1) delta-CRL [28], (2) Merkle hash tree (MHT) [27], and (3) one-way hash chain [29]. By combining these three mechanisms, we designed BECSI which allows increasing the availability and freshness of the certificate status information and at the same time reduces the bandwidth necessary to check the validity of a given certificate. BECSI takes advantage of V2V communication to create mobile repositories so that vehicles do not have to rely solely in the RSUs to obtain CSI. Therefore, BECSI reduces the peak bandwidth load associated with the CSI requests as there are more entities in the network that can answer these requests. To achieve, we combine the issuance of delta-CRLs and MHT.

By using the underlying concept of delta-CRLs, we implemented a more efficient way of distributing CSI inside the VANET. To help minimize frequent downloads of lengthy CRLs, delta-CRLs are published aperiodically. On the other hand, BECSI codes the information included in the CRL and delta-CRLs in different MHTs. As in COACH using this tree, any entity possessing the *extended-CRL* can act as repository and efficiently answer to certificate status checking requests of other vehicles. To evaluate the performance of BECSI we define three different metrics:

- *Query Cost (Q_{cost})*: This cost represents bandwidth requirement from repositories to vehicles. As it is shown in Fig. 2.11 while the CRL size grows

linearly with the number of revoked certificates, BECSI response sizes describe a logarithmic growth. Therefore, in terms of Query Cost, BECSI is more efficient than CRL and the compressed CRL and similar to ADOPT.

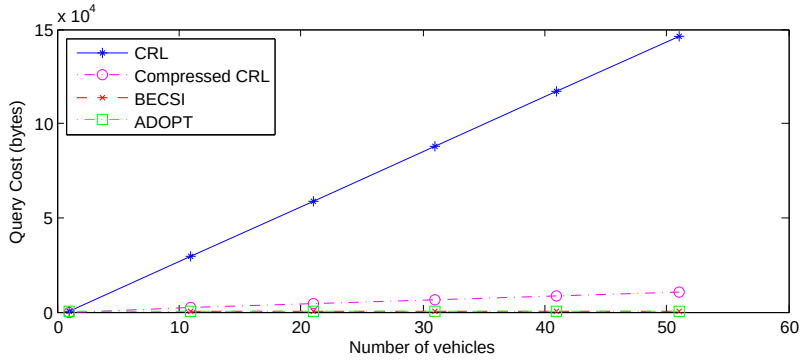


Figure 2.11: Response size vs. number of vehicles.

- *Request Ratio*: This metric captures the amount of requests that the entities perform to update the CSI. Figure 2.12 shows that BECSI and ADOPT have an almost constant request rate. Traditional schemes like CRL have lower request rates as there is only one request per validity period.

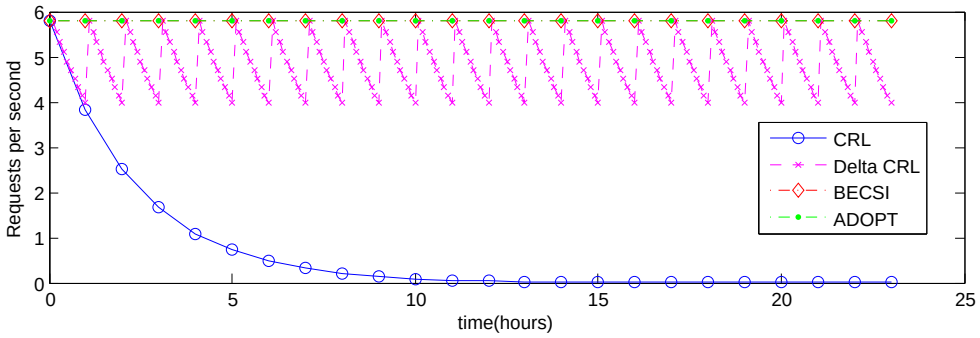


Figure 2.12: Request rate for different revocation mechanisms.

- *Window of vulnerability (WOV)*: This criterion captures the risk of operating with cached CSI. Figure 2.13 shows that CRL is the worst mechanism in terms of WOV. BECSI inherits the improvement in terms of WOV from the delta-CRL mechanism. Thus, BECSI highly improves traditional CRL in terms of WOV.

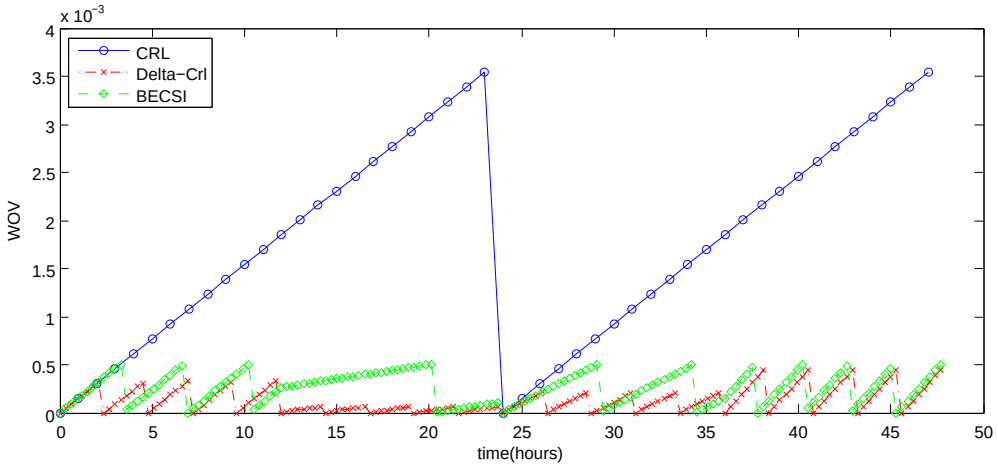


Figure 2.13: WOV for different revocation mechanisms.

Using the NCTUns [30] simulator we analyzed the performance of BECSI in a vehicular scenario. In VANETs, the most important issue in any revocation method is the delay of delivering the CSI to the vehicles to prevent that misbehaving vehicles from jeopardizing the safety of its neighbors. Consequently, we measured the revocation delay as delay from the moment a vehicle issues a CSI request until the moment the new CSI is received. Table 2.5 shows the average time spent by a vehicle to retrieve CSI from a repository.

Revocation Mechanism	**Average Time**	**Standard Deviation**
CRL (300 KB)	2,23 min	0,51 min
Compressed-CRL (20 KB)	7,01 sec	1,12 sec
Traditional Delta CRL (2.5-15 KB)	4,47 sec	2,12 sec
ADOPT (652 B)	705,06 ms	200,81 ms
BECSI Delta CRL (8 KB)	6,02 sec	0,05 sec
BECSI MHT (778 B-912 B)	483,02 ms	20,31 ms

Table 2.5: Time required to retrieve CSI.

It is worth noting that the worst mechanisms in terms of delay are the traditional CRL and delta-CRL as requesting entities are downloading all the available

CSI. However, the delay of the conventional CRL compared with the proposed BECSI protocol decreases with the number of CSI requests. The variations in time to download the CRL are due to the number of intermediate RSUs existing in the connection between the CA and the vehicle sending the revocation request. The average time to validate the status of a certificate in ADOPT is lower than BECSI because of the number of hops that are necessary to retrieve the cached CSI. BECSI in its MHT mode of operation is the fastest in average when validating the status of a certificate. However, this mode of operation has also a notable deviation. While in ADOPT the high deviation is due to the number of hops, in BECSI this deviation is mainly due to the number of Δ-trees that a vehicle has to check when a certificate is not revoked. Figure 2.14 shows the number of vehicles that are able to download the CSI in a particular range time depending on the revocation mechanisms. As expected, with BECSI and ADOPT almost all the 100 vehicles are able to download and process the CSI in less than 1,5 seconds. However, with Delta-CRLs and compressed-CRLs it takes from 4 to 8 seconds to retrieve the CSI.

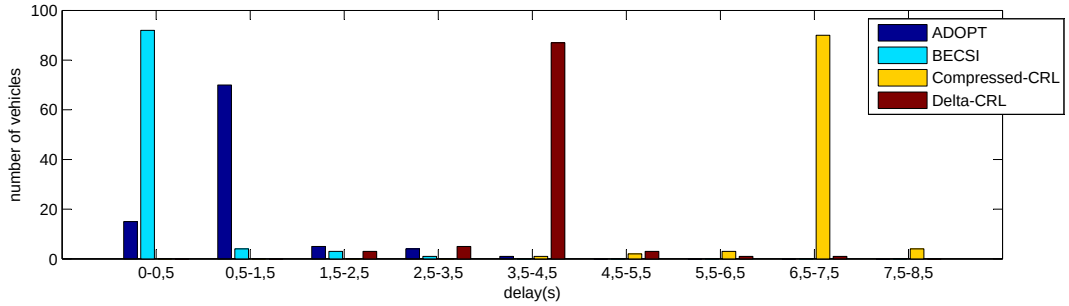


Figure 2.14: Histogram plot of time delay of the vehicles that receive the CSI depending on the revocation mechanism.

Thus, BECSI evaluation shows that it not only improves in terms of bandwidth but also in terms of scalability (increases the number of available repositories) and vulnerability (controlled WOV). In this way, BECSI becomes an offline certificate status validation mechanism as it does not need trusted responders to operate. Therefore, BECSI significantly achieves great efficiency and scalability, especially when deployed in heterogeneous vehicular networks.

2.4 Impact of the revocation service in PKI prices

As final part of this thesis, we analyzed the economic impact of the revocation service in the certificate prices. We noticed that with the appearance of novel network environments (e.g. VANET), the quantity of CAs in the SSL certificate market will become larger and the market concentration will diminish, but this will not simple eliminate the oligopoly in the short-term. During the 90s, the certification market, the competition among CAs appears mainly as price competition. In this situation, malignant price competition would be detrimental to the interests of the users and lead to the CA's pay crisis. Facing the situation, the main CAs began to change the competitive strategies from basic price competition to price and quality of services (QoS) competition. To provide better QoS, CAs have to improve their revocation service, and specifically the freshness of the CRLs. Users will pay more for a service that issues certificate status information faster. Time-to-revocation metric is visible to costumers by checking the CA's repositories where they publicize the revocation information.

We proved that there exists an oligopoly of CAs which compete in certificate prices and QoS. We assume that the revocation probability is *ex-ante* uncertain which is quite logical and intuitive. The number of revoked certificates varies with time and in a manner that cannot be predicted with certainty. We showed that an uncertain revocation probability introduces a systematic risk that does not decrease by selling more certificates. If CAs are risk averse, this effect relaxes price competition. The equilibrium characteristic of the certification market was found by establishing a price competition model with different QoS.

Firstly, we defined a utility function. We maximized this utility, assuming that the total utility U which users can get after they purchase a certificate consists of two parts. The first part is wealth utility which represented by U_w the other part is QoS utility which the applicant can get after they obtained the CA's services, represented by U_{QoS}. The total utility U is defined as:

$$U = \alpha_1 U_w + \alpha_2 U_{QoS}, \forall \alpha_k \in [0,1] \text{ and } \alpha_1 + \alpha_2 = 1. \tag{2.5}$$

where α_i represents the significance level of U respectively. We assume that the certification market is covered in full. Users will intend to maximize their utility,

i.e.:

$$\theta^* = \arg\max_{\theta} \quad U. \tag{2.6}$$

We obtained the certificate price and the coverage in the equilibrium, which allowed us to conclude that:

- In the equilibrium, two CAs with different revocation services achieve their maximum gain, the CA with better revocation service obtains a higher price for their certificates. This is mainly due to the fact that as both CAs have associated the same probability of being compromised, but the QoS of the first CA is better, this CA can set a higher price per certificate.
- In the equilibrium, the coverage that each CA should establish is the same and is inversely proportional to the risk-aversion and the probability of operating with a revoked certificate.

Finally, to corroborate the benefits of the presented model, we analyzed the case of current SSL providers that issue digital certificates. An SSL certificate can be obtained from amounts as low as $43 to as high as $3000 per year. Whilst the type of encryption can be the same, the cost is determined by the rigor of the certification process as well as the assurance and warranty that the vendor can provide. Table 2.6 shows the prices and QoS that the leading CAs operating in the SSL Certificate market are offering. The SSL Certificate market was traditionally dominated by a small number of players, namely VeriSign and Thawte. Whilst in a monopolistic position they had the capability of charging inflated prices for a commodity product. However new providers with no necessity to hold prices high were able to offer SSL certificates at far more reasonable prices.

To test whether these factors are determinant factors for the certificate prices, we perform a multivariate regression analysis explaining the yearly price of SSL certificates. General regression investigates and models the relationship between a response (Certificate price) and predictors (Warranty, issuing interval and CRL lifetime). Note that the response of this model is continuous, but we have both continuous and categorical predictors. With this model we determine how the certificate price changes as a particular predictor variable changes. We use data from a survey of CAs performed in 2010 [1]. The obtained regression model

is expressed in the following equations for high and low assurance certificates, respectively:

$$Price/Year(\$) = 98,4353 + 0,000220857\ W - 0,549141\ \overline{I_{time}} + 8,6116\ \frac{1}{\overline{CRL_{Lf}}},$$

$$Price/Year(\$) = 20,0405 + 0,000220857\ W - 0,5491411\ \overline{I_{time}} + 8,6116\ \frac{1}{\overline{CRL_{Lf}}},$$

where W denotes the warranty, $\overline{I_{time}}$ is the mean issuing time, and $\overline{CRL_{Lf}}$ is the mean lifetime of the CRLs issued by the CA.

Both regression equations show that the coefficient of the predictor associated to the CRL's mean lifetime is significant. Thus, it is demonstrated that the revocation service plays an important role when establishing the certificate prices.

SSL Provider	Product Name	Price/Year($)	Warranty($)	Assurance	Issuing time	Mean CRL lifetime
COMODO	EnterpriseSSL Platinum	311.80	1,000,000	High	<1 hour	4 days
COMODO	InstantSSL Pro	169.80	100,000	High	<1 hour	4 days
Verisign	Secure Site Pro Cert	826.67	2,500,000	High	2-3 days	15 days
Verisign	Managed PKI for SSL Std	234.00	100,000	High	2-3 days	15 days
GeoTrust	QuickSSL Premium	118.00	100,000	Low	Immediate	10 days
GeoTrust	True BusinessID	159.20	100,000	High	2 days	10 days
Go Daddy	Standard SSL	42.99	10,000	Low	Immediate	1 day
Go Daddy	Standard Wildcard	179.99	10,000	Low	Immediate	1 day
Entrust	Advantage SSL Certificates	167.00	10,000	High	2 days	1 week
Entrust	Standard SSL Certificates	132.00	10,000	High	2 days	1 week
Thawte	SSL 123	129.80	-	Low	Immediate	1 month
Thawte	SGC Super cert	599.80	-	High	2 days	1 month

Table 2.6: SSL Certificate Types and Services offered by main CAs [1].

Chapter 3

Quality Indexes

The research presented in this thesis has been validated internationally in various journals and conferences where experts have provided valuable comments and insights that have improved our research. Tables 3.1 and and 3.2 show the publications made during the development of the thesis In both tables, the displayed information gives evidence of the quality of each of them.

Year	Publication Title	Journal	Quality Index
2013	BECSI : Bandwidth Efficient Certificate Status Information distribution mechanism for VANETs	Mobile Information Systems	Impact Factor (ISI) = 2.432 h-index= 23
2012	Risk based decision making for Public Key Infrastructure using fuzzy logic	International Journal of Innovative Computing, Information and Control	Impact Factor (ISI) = 1.667 h-index= 32
2012	A Modeling of Certificate Revocation and Its Application to Synthesis of Revocation Traces	IEEE Transactions on Information Forensics and Security	Impact Factor (ISI) = 1.340 h-index= 41
2012	COACH: COllaborative certificate stAtus CHecking mechanism for VANETs	Journal of Network and Computer Applications	Impact Factor (ISI) = 1.065 h-index= 28

Table 3.1: Quality Indexes of the articles published in journals.

Year	Title	Conference	Quality Index
2012	Impact of the Revocation Service in PKI Prices	Information and Communications Security (ICICS 2012)	CORE Ranking B h-index= 20
2012	On the Self-similarity Nature of the Revocation Data	in Information Security Conference (ISC 2012)	CORE Ranking B h-index= 16
2012	Toward Revocation Data Handling Efficiency in VANETs	in Communication Technologies for Vehicles (Nets4Cars 2012)	No CORE ranking h-index= 2
2009	PKIX certificate status in hybrid MANETs	Information Security Theory and Practice. Smart Devices, Pervasive Systems, and Ubiquitous Networks (WISTP 2009)	CORE Ranking C h-index= 10

Table 3.2: Quality Indexes of the articles presented at international conferences.

Chapter 4

Conclusions

In this thesis, we have analyzed the revocation process and proposed a set of mechanisms to provide an efficient revocation service for VANETs. Our results have shown that the proposed mechanisms can achieve the targeted security requirements. In addition, the detailed performance evaluation and security analysis have indicated that the proposed protocols are secure and efficient. The achievements accomplished in this thesis can be summarized as follows:

- We have analyzed real empirical data collected from the leading CAs. We have shown that the revocation process is statistically self-similar (irrespective of when data were collected during the 3-year period 2008-2011 or from which CA). Moreover, we have demonstrated that the degree of self-similarity, which can be measured in terms of the Hurst parameter H, is a function of the overall utilization of the revocation service and can be used for measuring the "burstiness" of the revocation process (i.e. the more bursts in the revocation process the higher H). Hence, leading CAs share similar Hurst parameters even though they operate in different market segments.

- The intermittent connectivity between the entities of vehicular networks and security infrastructure results in incomplete or outdated revocation information at the recipients of signed messages. This incomplete/outdated information puts the recipients in a dilemma while accepting messages signed using certificates that are not present in the CRLs at the On Board Unit. To

this respect, we have presented a new metric that quantifies the confidence the recipients can have while accepting messages signed using certificates that are not present in the CRLs at the OBU. Moreover, we have developed a systematic methodology to build a fuzzy system that models risk and assists the user in the decision making process related to certificate revocation. Our system not only considers the possibility of taking as valid a certificate that has been revoked but also other key risk factors. In this respect, we have identified potential risk sources involved in the revocation system and we have characterized them using fuzzy logic. The inputs given to the fuzzy system can be inferred from a standard CRL and as CRLs are accessible to any PKI user, in practice, everybody can take advantage of our fuzzy system. The output of our system is a measure of the risk of operating with a particular CRL at a given instant. Based on this output, users can either decide whether to trust or not a given signed message.

- We have proposed two efficient revocation mechanism for VANETs, which substantially reduce the overhead of the certificate status checking. These checking mechanisms are based on an *extended-CRL*. The main advantage of this *extended-CRL* is that the road-side units and repository vehicles can build an efficient structure based on an authenticated hash tree to respond to status checking requests inside the VANET, saving time and bandwidth. Thus, we decrease the vulnerability window that a misbehaving vehicle has and this results in higher safety level for VANET. Both mechanisms are resistant to the most known revocation attacks. In addition, they can be efficiently integrated with any PKI and/or any misbehavior detection scheme for VANETs and they fulfill the IEEE 1609.2 Standard.

- Finally, we have studied the economic impact of the revocation service in the certificate price. We have shown that the market of certificate providers can be described as an oligopoly where oligarchs compete not only in price but also in quality of service. We have modeled this oligopoly using a game theoretic approach to find the prices in the equilibrium. We have been able to capture the QoS of the products offered by a CA, by means of the timeliness of the revocation mechanism and the security level. In

our model of the certification industry with profit-maximizing CAs and a continuum of individuals, we showed that although the undercutting process in certification prices seems similar to the price setting behavior of firms in Bertrand competition there exists a crucial difference depending on the QoS of the revocation service. The solution of the game for two CAs in the oligopoly that offer certificates with different QoS shows that the revenues of the CA which provides a better revocation mechanism and a higher security level are larger. Therefore, a CA has to take into account not only the probability of operating with a revoked certificate, but also the quality of the revocation mechanism and the security level when setting the prices of its certificates and the compensation expenses. Thus, any CA should comprehensively consider the difference in quality of its services compared with other CAs.

Appendix A

Publications

[1] C. Gañán, J. L. Muñoz, O. Esparza, J. Mata-Díaz, J. Hernández-Serrano, and J. Alins, "COACH: COllaborative certificate stAtus CHecking mechanism for VANETs," *Journal of Network and Computer Applications*, Mar. 2012. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1084804512000616

[2] C. Gañan, J. L. Muñoz, O. Esparza, J. Loo, J. Mata-Díaz, and J. Alins, "BECSI : Bandwidth Efficient Certificate Status Information distribution mechanism for VANETs," *Mobile Information Systems*, pp. 1–31, 2013, (in press). [Online]. Available: http://dx.doi.org/10.3233/MIS-130167

[3] C. Gañán, J. Mata-Diaz, J. L. Munoz, J. Hernandez-Serrano, O. Esparza, and J. Alins, "A Modeling of Certificate Revocation and Its Application to Synthesis of Revocation Traces," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 6, pp. 1673–1686, Dec. 2012. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6247505

[4] C. Gañán, J. Muñoz, O. Esparza, J. Mata-Díaz, and J. Alins, "Risk-based decision making for Public Key Infrastructure using fuzzy logic," *International Journal of Innovative Computing, Information and Control (IJICIC)*, vol. 8, no. 11, pp. 7925–7942, 2012. [Online]. Available: http://www.ijicic.org/ijicic-ksi-07.pdf

[5] C. Gañán, J. L. Muñoz, O. Esparza, J. Mata-Díaz, and J. Alins, "Impact of the Revocation Service in PKI Prices," in *Information and Communications Security*, ser. Lecture Notes in Computer Science, T. Chim and T. Yuen, Eds., vol. 7618. Hong Kong: Springer Berlin Heidelberg, 2012, pp. 22–32. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-34129-8_3

[6] C. Gañán, J. Mata-Díaz, J. L. Muñoz, O. Esparza, and J. Alins, "On the Self-similarity Nature of the Revocation Data," in *Information Security*, ser. Lecture Notes in Computer Science, D. Gollmann and F. Freiling, Eds., vol. 7483. Passau: Springer Berlin Heidelberg, 2012, pp. 387–400. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-33383-5_24

[7] J. Muñoz, O. Esparza, C. Gañán, and J. Parra-Arnau, "PKIX certificate status in hybrid MANETs," in *Information Security Theory and Practice. Smart Devices, Pervasive Systems, and Ubiquitous Networks*, ser. Lecture Notes in Computer Science, O. Markowitch, A. Bilas, J.-H. Hoepman, C. Mitchell, and J.-J. Quisquater, Eds. Springer Berlin Heidelberg, 2009, vol. 5746, pp. 153–166. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-03944-7_12

[8] C. Gañán, J. Muñoz, O. Esparza, J. Mata-Díaz, and J. Alins, "Toward Revocation Data Handling Efficiency in VANETs," in *Communication Technologies for Vehicles*, ser. Lecture Notes in Computer Science, A. Vinel, R. Mehmood, M. Berbineau, C. Garcia, C.-M. Huang, and N. Chilamkurti, Eds., vol. 7266. Vilnius: Springer Berlin Heidelberg, 2012, pp. 80–90. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-29667-3_7

ATTENTION ¡

Pages 39 to 85 of the thesis containing publications 1, 2, and 3 are available at the editor's web

BECSI: Bandwidth Efficient Certificate Status Information distribution mechanism for VANETs

Carlos Gañán*[1], Jose L. Muñoz[1], Oscar Esparza[1], Jonathan Loo[2], Jorge Mata-Díaz[1], and Juanjo Alins[1]

[1] *Telematics Department, Universitat Politècnica de Catalunya, Barcelona, Spain*
[2] *Computer Communications Department, Middlesex University, London, UK.*

Abstract

Certificate revocation is a challenging task, especially in mobile network environments such as vehicular ad Hoc networks (VANETs). According to the IEEE 1609.2 security standard for VANETs, public key infrastructure (PKI) will provide this functionality by means of certificate revocation lists (CRLs). When a certificate authority (CA) needs to revoke a certificate, it globally distributes CRLs. These lists must be distributed as quickly and efficiently as possible without over-burdening the network. In this article, we propose BECSI, a Bandwidth Efficient Certificate Status Information mechanism to efficiently distribute certificate status information (CSI) in VANETs. By means of Merkle hash trees (MHT), BECSI allows to retrieve authenticated CSI not only from the infrastructure but also from vehicles acting as mobile repositories. Since these MHTs are significantly smaller than the CRLs, BECSI reduces the load on the CSI repositories and improves the response time for the vehicles. Additionally, BECSI improves the freshness of the CSI by combining the use of delta-CRLs with MHTs. Thus, vehicles that have cached the most current CRL can download delta-CRLs to have a complete list of revoked certificates. Once a vehicle has the whole list of revoked certificates, it can act as mobile repository.

Keywords: PKI, Revocation, VANET.

*Corresponding author: Carlos Gañán, Universitat Politècnica de Catalunya (UPC), Jordi Girona 1-3, 08034 Barcelona, Spain. E-mail: carlos.ganan@entel.upc.edu, Phone: +34 93 401 7027, Fax: +34 93 401 1058

1 Introduction

Vehicular ad-hoc networks (VANETs) have recently attracted extensive attentions as a promising technology for revolutionizing the transportation systems. VANETs consist of entities including On-Board Units (OBUs) and infrastructure Road-Side Units (RSUs). Mobile nodes are capable of communicating with each other (i.e. Vehicle to Vehicle Communication -V2V communication) and with the RSUs (i.e. Vehicle to Infrastructure Communication -V2I communication). Multi-hop communication facilitates information exchange among network nodes that are not in direct communication range [3, 14], by means of short range wireless technology based on IEEE 802.11p.

Obviously, any malicious behaviors, such as injecting beacons with false information, modifying and replaying the previously disseminated messages, could be fatal to the other users. Thus, identifying the message issuer is mandatory to reduce the risk of such attacks. According to the IEEE 1609.2 standard [13], vehicular networks will rely on the public key infrastructure (PKI). In PKI, a certification authority issues an authentic digital certificate for each node in the network. Due to misbehavior, intentional or otherwise, certificates need to be revoked in order to limit the risk that potential misuse poses to the rest of the network. The IEEE 1609.2 standard [13] states that VANETs will depend on certificate revocation lists (CRLs) to achieve revocation. CRLs are black lists that enumerate revoked certificates along with the date of revocation and, optionally, the reasons for revocation.

As VANETs can have a great amount of nodes (i.e. vehicles), CRLs will be large. Moreover, each vehicle in the network will own many temporary certificates (also called pseudonyms) to protect the users' privacy. Consequently, these lists will require hundreds of Megabytes [12, 21, 35]. However, distributing and updating CRLs to all vehicles raises a challenge. If there are no more communication media than the own VANET, no trusted-third parties (like the corresponding CA) can be assumed to be permanently available. Thus, online certificate status protocol (OCSP) [28] or, in general, any online solution is not suitable for this context. Several CRLs distribution protocols have been proposed for this purpose. For instance, to distribute these lists efficiently, authors in [26] proposed revocation using compressed CRLs. They divided the CRL into several self-verifiable parts and strongly reduced its size by using Bloom filters. Authors in [12] also propose the use of Bloom filters to store the revoked certificates for increasing the search speed in the CRL. On the other hand, authors in [22] proposed to use regional CAs and short lived certificates to

decrease the number of entries in the CRL. We provide more information about these and other similar proposals in Section 2 but as a general conclusion, we could say that most of the research efforts in this context have been put on trying to reduce the size of the CRL, either trying to split it or trying to compress it.

In this article, we address the CRL distribution problem by exploiting the combination of three well-known mechanisms: (1) delta-CRL [1], (2) Merkle hash tree (MHT) [18], and (3) one-way hash chain [16]. By combining these three mechanisms, we design a Bandwidth Efficient Certificate Status Information (BECSI) protocol, that allows increasing the availability and freshness of the certificate status information (CSI) and at the same time reduces the bandwidth necessary to check the validity of a given certificate. BECSI takes advantage of V2V communication to create mobile repositories so that vehicles do not have to rely solely in the RSUs to obtain CSI. We aim to improve the distribution of CSI by transmitting the revocation information that is unknown to a particular user during the validity period of a CRL. The main idea behind BECSI is to allow vehicles requesting for new CSI during the validity period of the current CRL. Thus, revocations that occur during the validity period of the CRL will not be unknown to the vehicles during this whole validity period, reducing the risk of operating with an unknown revoked certificate. Therefore, BECSI reduces the peak bandwidth load associated with the CSI requests as there are more entities in the network that can answer these requests. To achieve, we combine the issuance of delta-CRLs and MHT.

By using the underlying concept of delta-CRLs, we implement a more efficient way of using of distributing CSI inside the VANET. To help minimize frequent downloads of lengthy CRLs, delta-CRLs are published aperiodically. On the other hand, BECSI codes the information included in the CRL and delta-CRLs in different MHTs. Using these MHTs, vehicles are able to act as mobile repositories. To achieve that, we embed some little extra information to the CRL such that allows us to create an efficient and secure request/response protocol. In more detail, we propose a way of efficiently embedding a MHT within the structure of the standard CRL to generate the so-called *extended-CRL* and *extended-delta-CRL*. To create these extended lists, we use an standard way of adding extra information to the CRL. Our extension contains all the necessary information to allow any vehicle or VANET infrastructure element that possesses the *extended-CRL* to build the BECSI tree, i.e., a hash tree with the CSI of the CRL. Using this BECSI tree, any entity possessing the *extended-CRL* can act as repository and efficiently answer to certificate status checking requests of other vehicles. As we will demonstrate by simulation, this makes the distri-

bution of CSI more efficient than distributing complete CRLs (even they are compressed), reducing the data that have to be transmitted over the VANET. We must stress that any entity possessing an *extended-CRL* can act as BECSI repository but that a BECSI repository is not a TTP. In other words, BECSI is offline, which means that no online trusted entity (like a CA) is needed for authenticating the responses produced by BECSI repositories.

The rest of this paper is organized as follows. In Section 2, we present the background related to our mechanism. In Section 3 we describe in depth BECSI. In Section 4, we evaluate the proposed mechanisms. Finally, Section 5 concludes this paper.

2 Background

In this section, first we start describing existing revocation proposals for VANET. Then, we give a brief overview of Merkle Hash Trees (MHT) [18], which is one of the foundations of the proposed certificate validation mechanism. Finally we describe the basics of hash chains.

2.1 VANET revocation mechanisms

2.1.1 Centralized revocation approaches

The IEEE 1609.2 standard [13] proposes an architecture based on the existence of a Trusted Third Party (TTP), which manages the revocation service. In this architecture each vehicle possesses several short-lived certificates (used as pseudonyms), to ensure users' privacy. However, short-lived certificates are not enough as compromised or faulty vehicles could still endanger other vehicles until the end of their certificate lifetimes. Thus, the IEEE 1609.2 promotes the use of CRLs to manage revocation while assuming pervasive roadside architecture.

Other proposals in the literature also assume the existence of a TTP to provide the revocation service. Raya *et al.* [25] propose the use of a tamper-proof device (TPD) to store the certificates. A TTP is in charge of preloading the cryptographic material in the TPD. Thus, when a vehicle is compromised/misbehaving, it can be removed from the network by just revoking the TPD. To ensure that messages from this OBU are not considered valid once the certificates have been revoked, revocation information must also be distributed via CRLs. To reduce the bandwidth consumed by the transmission of CRLs, these authors proposed to compress the CRLs by using Bloom filters. However,

this method gives rise to false positives which degrades the reliability of the revocation service.

However, even compressed, timely distributing CRLs to all vehicles is not trivial. Some authors [22, 23], instead of using a single central authority, have proposed the use of regional certification authorities which must develop some trust relationships. Papadimitratos *et al.* [24] suggest restricting the scope of the CRL within a region. Visiting vehicles from other regions require to obtain temporary certificates. Thus, a vehicle will have to acquire temporary certificates if it is traveling outside its registered region. The authors also propose breaking the Certificate Revocation List (CRL) into different pieces, then transmitting these pieces using Fountain or Erasure codes, so that a vehicle can reconstruct the CRL after receiving a certain number of pieces. Similarly, in [32], each CA distributes the CRL to the RSUs in its domain through Ethernet. Then, the RSUs broadcast the new CRL to all the vehicles in that domain. In the case RSUs do not completely cover the domain of a CA, V2V communications are used to distribute the CRL to all the vehicles [15]. This mechanism is also used in [2, 7], where it is detailed a public key infrastructure mechanism based on bilinear mapping. Revocation is accomplished through the distribution of CRL that is stored by each user.

2.1.2 Decentralized revocation approaches

Decentralized revocation mechanisms provide the revocation service without assuming the existence of a TTP. Some proposals in the literature divert from the IEEE 1609.2 standard and use online status checking protocols instead of CRLs to provide a revocation service in a decentralized manner. This is the case, of the Ad-hoc Distributed OCSP for Trust (ADOPT) [17], which uses cached OCSP responses that are distributed and stored on intermediate nodes. Other group of proposals bases the revocation service on detecting a vehicle to be misbehaving by a set of other vehicles. Then, the detecting set may cooperatively revoke the credential of the misbehaving node from their neighborhood. Moore *et al.* proposed in [19] a revocation mechanism aiming to prevent an attacker from falsely voting against legitimate nodes. Raya *et al.* in [25] proposed a mechanism to temporarily revoke an attacker if the CA is unavailable. To do so, the number of accusing neighbor users must exceed a threshold. A similar mechanism based also on vehicle voting is proposed in [34]. Again, by means of a voting scheme, a vehicle can be marked as misbehaving and then be revoked by its neighbors.

Another proposal uses a game-theoretic revocation approach to define the

best strategy for each individual vehicle [5,27]. These mechanisms provide incentives to guarantee the successful revocation of the malicious nodes. Moreover, thanks to the records of past behavior, the mechanism is able to dynamically adapt the parameters to nodes' reputations and establish the optimal Nash equilibrium on-the-fly, minimizing the cost of the revocation.

Finally, there are some hybrid approaches that are neither totally centralized nor decentralized ([10, 33]). For instance, authors in [9] propose the use of authenticated data structures to issue CSI. Using these schemes, the revocation service is decentralized to transmit the CSI but still depends on a CA to decide when a node should be evicted from the VANET.

2.2 The Merkle Hash Tree

A Merkle hash tree (MHT) [18] is essentially a tree structure that is built with a One Way Hash Function (OWHF). The leaf nodes hold the hash values of the data of interest (`data1`,`data2`,...) and the internal nodes hold the hash values that result from applying the OWHF to the concatenation of the hash values of its children nodes. In this way, a large number of separate data can be tied to a single hash value: the hash at the root node of the tree. MHTs can be used to provide an efficient and highly-scalable way to distribute revocation information, as it is described in [8] for MANETs (Mobile Ad Hoc Networks). A sample MHT is presented in Fig. 1. This hash tree is binary because each node has at most two children or equivalently, two sibling nodes are combined to form a parent node in the next level. We will call these siblings as "left" and "right" and a detailed explanation of how to build the hash tree for BECSI is given in Section 3.3.

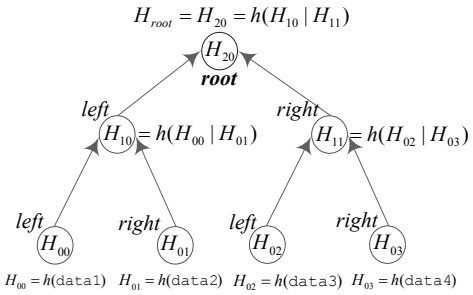


Figure 1: Sample binary Merkle Hash Tree.

A MHT relies on the properties of the One Way Hash Functions (OWHF). It exploits the fact that an OWHF is at least 10,000 times faster to compute than a digital signature, so the majority of the cryptographic operations performed in the revocation system are hash functions instead of digital signatures. In addition, by storing the internal node values, it is possible to verify that any of the leaf nodes is part of the tree without revealing any of the other data.

2.3 Hash chains

The idea of "hash chain" was first proposed by Lamport [16] in 1981 and suggested to be used for safeguarding against password eavesdropping. A hash chain $\mathcal{C}$ is a set of values $s_0, ..., s_n$ for $n \in \mathbb{Z}$ such that $s_i = h(s_{i-1})$ for some one-way hash function h, where $i \in [1, n]$ and s_0 is a valid input for h.

Note that hash chains are preimage resistent, i.e., by knowing s_i, s_{i-1} cannot be generated by those who do not know the value s_0, however given s_{i-1}, its correctness can be verified by hashing $h(s_i)$. This property of hash chains has evolved from the property of one-way hash functions. Additionally, hash chains are also second preimage resistent, collision resistent and generate pseudo-random numbers.

In most of the hash-chain applications, first s_n is securely distributed and then the elements of the hash chain are spent (or used) one by one by starting from s_{i-1} and continuing until the value of s_0 is reached. At this point the hash chain is said to be exhausted and the whole process should be repeated again with a different s to reinitialize the systems.

3 BECSI: Bandwidth Efficient Certificate Status Information distribution mechanism

In this Section, we present BECSI, a bandwidth efficient mechanism for certificate status checking over VANETs based on the use of Merkle Hash Trees and hash chains. First we introduce the motivation, goal and security architecture needed to support BECSI, and next we describe the mechanism in depth.

3.1 Motivation and Goal

Despite the short-comings related to propagation of revocation information, the need for trusted authorities like CAs to ensure authentication has motivated

researchers to propose PKI based security for vehicular networks. Mainly, these mechanisms intend to provide the following set of requirements:

1. *Reliability*: The revocation service must be available at all times.
2. *Memory*: Minimum amount of memory should be required as validation is often carried out in constrained environments.
3. *Bandwidth*: Communication bandwidth should be minimal.
4. *Freshness*: Revocation data should be as updated as possible.

Proposals described in Section 2 mainly deal with the bandwidth requirement. By compressing the CRL, using state-of-the-art coding techniques or partitioning the CRL, these approaches reduce the time required to download CSI. In addition, authors intend to provide a reliable revocation service by decentralizing the CSI distribution points. However, none of these works deals with the freshness of the CSI. With BECSI we aim not only to reduce the communication overhead but also to increase the availability and freshness of CSI while keeping a reasonable computation cost.

CRLs are normally published in intervals meaning that there will not be any new revocation information available between the issuance and the update of the CRL. Newer revocations will thus be delayed until the next update occurs. High-security applications (e.g. safety applications) cannot cope with this lack of fresh information and render the traditional CRL approach almost useless in VANETs. To solve these problems, BECSI includes an extension to the standard CRL that allow RSUs to act as an offline repositories. Thus vehicles do not have to download the whole extended CRL, and they can just query about the status of a particular certificate.

3.2 Security Architecture

The security architecture is an adaptation of a mesh PKI system to a vehicular scenario constructed of peer-to-peer CA relationships. This architecture consist of 4 different types of nodes (see Fig. 2):

1. *Certification Authorities*: CAs are responsible for holding and managing the credentials and identities of all the vehicles which are registered under its hood. CAs are responsible for generating the set of certificates that are stored in each OBU. They are also responsible for managing the revocation information and making it accessible to the rest of the entities. By

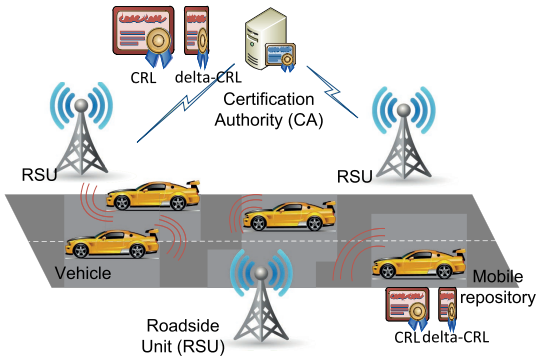


Figure 2: System Architecture.

definition of TTP, the CA should be considered fully trusted by all the network entities, so it should be assumed that it cannot be compromised by any attacker. In fact, in our proposal CAs are the only trusted entities within the network.

2. *Road-Side Units*: RSUs are fixed entities that are fully controlled by the CA. They can access the CA anytime because they are located in the infrastructure-side, which does not suffer from disconnections. If the CA considers that an RSU has been compromised, the CA can revoke it.

3. *Vehicles*: They are the clients of the network. They have their cryptographic material stored in a TPD. Vehicles can check the validity of a certificate using V2I or V2V.

4. *Mobile Repositories*: Mobile repositories are vehicles that have previously downloaded the CRL/delta-CRLs and are willing to response to cerficate status requests from other vehicles.

3.3 BECSI Tree

In this section, we introduce the data structure that BECSI uses to handle the revocation service. In this sense, we define the BECSI tree as a composite Merkle Hash Tree (see section 2). This tree consists of:

- A *base*-tree which is constructed using the serial number of the revoked certificates contained in the base-CRL.

- A set of Δ-trees which are constructed from the serial number of the certificates that are revoked during the validity interval of the base-CRL, i.e, they are constructed from the data contained in the delta-CRLs.

3.3.1 BECSI *base*-tree

The *base*-tree is a binary hash tree where each node represents a revoked certificate that is contained in the base-CRL. We denote by $N_{i,j}$ the nodes within the BECSI *base*-tree, where $i, j \in \{0, 1, 2...\}$ represent respectively the i-th level and the j-th node in the i-th level. We denote by $H_{i,j}$ the cryptographic (hash) value stored by node $N_{i,j}$ (see Fig. 3).

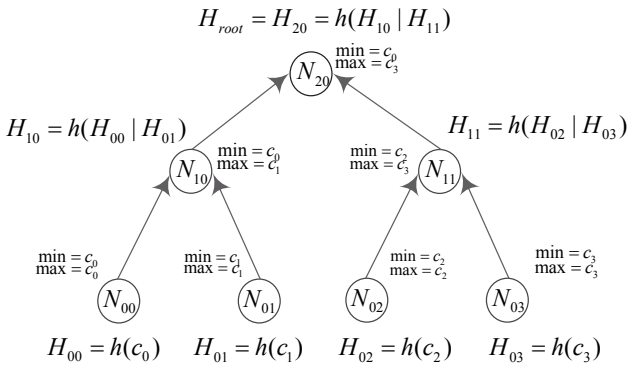


Figure 3: Sample BECSI *base*-tree.

We denote by $N_{i,j}$ the nodes within the MHT where i and j represent respectively the i-th level and the j-th node. We denote by $H_{i,j}$ the cryptographic variable stored by node $N_{i,j}$.

Nodes at level 0 are called "leaves" and they represent the data stored in the tree. In the case of revocation, leaves represent the set Φ of certificates that have been revoked,

$$\Phi = \{c_0, c_1, \ldots, c_j, \ldots, c_n\}\,. \tag{1}$$

Where c_j is the data stored by leaf $N_{0,j}$. Then, $H_{0,j}$ is computed as:

$$H_{0,j} = h(c_j)\,, \tag{2}$$

where h is a OWHF.

To build the MHT, a set of t adjacent nodes at a given level i ($N_{i,j}$, $N_{i,j+1}$, ..., $N_{i,j+t-1}$) are combined into one node in the upper level, which we denote

by $N_{i+1,k}$. Then, $H_{i+1,k}$ is obtained by applying h to the concatenation of the t cryptographic variables:

$$H_{i+1,k} = h(H_{i,j} | H_{i,j+1} | \ldots | H_{i,j+t-1}). \quad (3)$$

At the top level there is only one node called the "root". H_{root} is a digest for all the data stored in the MHT.

The sample MHT of Fig. 1 is a binary tree because adjacent nodes are combined in pairs to form a node in the next level ($t = 2$) and $H_{root} = H_{2,0}$.

We define the the $\mathcal{D}$igest as the concatenation of the certification authority distinguished number, the root hash and the validity period of the certificate status data. Once created, the $\mathcal{D}$igest is signed by the CA.

$$\mathcal{D}\text{igest}_{base} = \{DN_{CA}, H_{root}, ValidityPeriod\}_{SIG_{CA}}.$$

We denote as the $\mathcal{P}\text{ath}_{c_j}$ as the set of cryptographic values necessary to compute H_{root} from the leaf c_j.

It is worth noting that $\mathcal{D}$igest is trusted data because it is signed by the CA and it is unique within the tree while $\mathcal{P}$ath is different for each leaf. Thus, If the MHT provides a response with the proper $\mathcal{P}\text{ath}_{c_j}$ and the MHT $\mathcal{D}$igest, any vehicle can verify whether $c_j \in \Phi$.

For instance, let us suppose that a certain user wants to find out whether c_1 belongs to the sample MHT of Fig. 1. Then,

$\mathcal{P}\text{ath}_{c1} = \{H_{0,0}, H_{1,1}\}$,

$\mathcal{D}igest = \{DN_{CA}, H_{2,0}, ValidityPeriod\}_{SIG_{CA}}$.

The response verification consists in checking that $H_{2,0}$ computed from the $\mathcal{P}_{c_1}$ matches $H_{2,0}$ included in the $\mathcal{D}$igest:

$$H_{root} = H_{2,0} = h(h(h(c_1)|H_{0,0})|H_{1,1}).$$

Note that the BECSI *base*-tree can be built by a trusted third party (e.g. a CA) and distributed to a non-TTP because a leaf cannot be added or deleted to Φ without modifying H_{root}, which is included in the $\mathcal{D}$igest and as the $\mathcal{D}$igest is signed, it cannot be forged by a non-TTP. To do such a thing, an attacker would need to find a pre-image of a OWHF which is computationally infeasible by definition.

3.3.2 BECSI Δ-trees

BECSI Δ-trees are constructed in the same way that the *base*-tree. However they present two differences with respect to the *base*-tree:

- Each leaf of the Δ-trees refers to certificates that were revoked during the validity interval of the base-CRL.
- The root of the Δ-tree is calculated by hashing the top-hash of the tree with the corresponding value of a hash chain. For more details about the construction of the hash chain see Section 3.4.3.

Fig. 4 shows the simplest possible Δ-tree which contains only two revoked certificates.

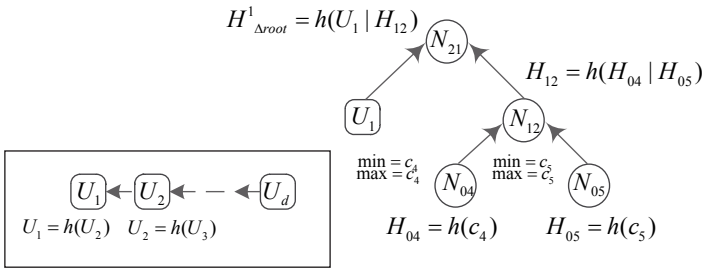


Figure 4: Sample BECSI Δ-tree.

Note that these Δ-trees have the same properties that the *base*-tree, so that the root node is unique and cannot be forged. Thus, the $\mathcal{D}$igest is composed as:

$$\mathcal{D}\text{igest}_{\Delta_i} = \{DN_{CA}, H^i_{\Delta root}, ValidityPeriod\}_{SIG_{CA}}.$$

Similarly, the $\mathcal{P}$ath consist of the set of cryptographic values necessary to compute $H^i_{\Delta root}$ from the leaf c_j. Note that this path is shorter in the case of the leafs of the Δ-trees because they contain less revoked certificates than the *base*-tree. The length of the Δ-trees is fixed as they are constructed from the delta-CRLs that have fixed size.

3.4 Operating Mode

BECSI consists in four phases. During the first phase of *Bootstraping*, the CA creates the "extended-CRL", that is, a CRL in which a signed extension is appended. This extension will allow non-trusted third parties (non-TTP) to answer CSI requests in an off-line way when required. Once this extended-CRL has been constructed, it is distributed to the RSUs. In the second phase of *Repository Creation*, a non-trusted entity (i.e. a RSU or a vehicle) gets the extended-CRL and becomes a CSI repository for other VANET entities. Next,

during the third phase *CSI Update*, the CA creates "extended-delta-CRLs" of fixed size. A delta-CRL is a time-stamped digitally signed revocation list containing information about new revocations that occurred since the issuance of a prior base-CRL. A base-CRL is a complete CRL that contains a complete list of revoked certificates, to which the revocation list in the delta-CRL needs to be applied to produce the latest list of revoked certificates. Base and delta CRLs have similar data structures. To construct these fixed-size delta-CRLs, the CA has to wait to have enough new revoked certificates. Therefore, the issuance of these delta-CRLs is aperiodic. Once, the delta-CRL is constructed, the CA appends a signed extension corresponding to the root node of the Δ-tree. The extended-delta-CRLs are distributed to the RSUs and mobile repositories. Moreover, the CA broadcasts the number of issued extended-delta-CRLs to avoid CSI suppression attacks. Finally, in the fourth stage of *Certificate Status Checking*, vehicles can use an efficient protocol to obtain the CSI from any available VANET repository. Henceforward, we give a more detailed description of these three stages.

3.4.1 Bootstraping

In this first phase, the CA creates the *extended-CRL* and delivers it to the RSUs. An *extended-CRL* is basically a standard CRL with an appended extension. This extension can be used by non-TTP (e.g. RSUs and vehicles inside the VANET) to act as repositories and answer to CSI requests. All the tasks of this system initialization are performed in the CA locally.

These are the steps that the CA must carry out:

1. The CA creates a *tbs-CRL* (to be signed CRL), i.e., a list including the serial number of the certificates that have been revoked (along with the date of revocation), the identity of the CA, some time-stamps to establish the validity period, etc.

2. The CA creates the BECSI *base*-tree, i.e., a MHT constructed with the serial numbers within the previous *tbs-CRL* as leaves of the tree. The BECSI *base*-tree is a binary tree, and is constructed following the methodology explained in Section 3.3. The leaves of the *base*-tree are ordered by increasing serial number. Therefore, the bottom left leaf stores the revoked certificate with lowest serial number. Note that if the BECSI base-tree is formed by an odd number n of leaves, there is a leaf $N_{0,n-1}$ that does not have a pair. Then the single node is simply carried forward to the upper level by hashing its $H_{0,n-1}$ value. We proceed in the same way if any i-th

level is formed by an odd number n of nodes. Once created the MHT, the CA obtains the root hash.

3. The CA calculates the extension, which consists basically of the $\mathcal{D}$igest and the first value U_0 of a hash chain. The hash chain will be used to make users aware of the number of issued delta-CRLs. Recall that the $\mathcal{D}$igest is calculated as the concatenation of the CA distinguished number, the root hash of the base-tree and the validity period of the CSI, and after that signed by the CA. Obviously, the distinguished number and the validity period should be the same than the ones contained in the tbs-CRL. In fact, the BECSI base-tree is just a different way of representing the CSI, but the hash tree will be valid during the same time and will provide the same information than the CRL. Once calculated, this $\mathcal{D}$igest is appended to the tbs-CRL.

4. The CA creates the hash chain. To that end, the CA picks a random value for U_d. By hashing this value iteratively, the CA forms a one-way chain of self-authenticating values, and assigns the values sequentially to the time intervals (one value per delta-CRL). The last value of the chain U_0 is appended to the tbs-CRL along with the $\mathcal{D}$igest, generating the *tbs-extended-CRL*.

5. The CA signs the tbs-extended-CRL, generating the *extended-CRL*. Notice that this second overall signature not only authenticates all the CSI, but also binds this CSI to the $\mathcal{D}$igest. The *extended-CRL* is only slightly larger than the standard CRL, as we will show later in Section 4.

6. Finally, the CA distributes copies of the *extended-CRL* to the designated RSUs, which will act as the typical PKI repositories, in the same manner as they would do with a standard CRL.

After this first phase, the RSUs have a copy of the *extended-CRL*, which contains exactly the same CSI than a standard CRL and it is valid for the same time. The advantage of an *extended-CRL* is that any non-TTP in possession of it can generate again the BECSI base-tree locally, and obtain the root hash. As the *extended-CRL* also includes the $\mathcal{D}$igest, which is signed by the CA, this entity has an authenticated version of the BECSI base-tree and can answer to CSI requests in an off-line way.

3.4.2 CSI Repositories creation

In this phase, RSUs and freewill vehicles become new CSI repositories of the VANET. Vehicles that become mobile repositories allow the distribution of CSI in areas with poor coverage. To become a repository an entity must follow the following steps:

1. The entity obtains the *extended-CRL* (and *extended-delta-CRLs*)either from the CA or from another entity that has an up-to-date copy of the *extended-CRL* (and *extended-delta-CRLs*) in its cache. Notice that the CA uses a secure wireline to communicate with the RSUs, while the RSUs use a wireless link to communicate with the vehicles.

2. Once the *extended-CRL* (and *extended-delta-CRLs*) has been downloaded, the entity verifies that the signature of the *extended-CRL* (and *extended-delta-CRLs*) is valid and corresponds to the CA. If so, the entity generates locally the BECSI base-tree (and Δ-trees) using the serial numbers within the *extended-CRL* (and *extended-delta-CRLs*) and following the same algorithm than the CA (as explained in Section 3.3). The root hash of the tree created from the *extended-CRL* (and *extended-delta-CRLs*) entries must match the signed root value contained in the $\mathcal{D}\text{igest}_{base}$ (and $\mathcal{D}\text{igest}_{\Delta_i}$).

3. At this moment, the entity can respond to any status checking request from any vehicle until the corresponding $\mathcal{D}$igest expires.

3.4.3 CSI Update

After the first two phases, any entity of the VANET is capable of downloading the CRL from a repository or it can just check the status of a given certificate using the capabilities of a MHT. However, in order to improve the freshness of the revocation information and avoid potential bottlenecks when obtaining new CSI, BECSI also provides CSI updates during the validity interval of the CRL.

To alleviate high CRL distribution costs, BECSI uses a hybrid delta-CRL scheme. BECSI issues a variable number of delta-CRLs during the validity interval of the base CRLs (as shown in Fig. 5), reducing the total bandwidth load on the CRL distribution points (RSU and mobile repositories). The size of these delta-CRLs is fixed a priori by the CA. Consequently, the number of delta-CRLs issued during the validity interval of the base-CRL depends on the number of revoked certificates during this interval.

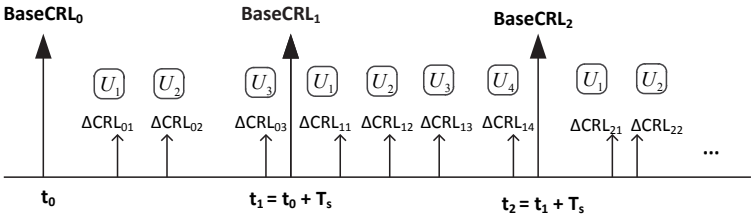


Figure 5: Delta-CRLs Issuance Scheduling.

To ensure that any vehicle entity is aware of the number of issued delta-CRLs, the CA discloses a value U_i of a hash chain each time a new delta-CRL is issued. This hash value allows users to make sure of how many Δ-trees have been published by the CA. Thus a non-TTP cannot lie about the amount of revocation information that has been published. Note that the corresponding value U_i is used to calculate the root value of the Δ-tree, binding the Δ-tree to the U_i. BECSI takes advantage of the physical layer used in VANETs to transmit the hash value U_i to vehicles.

The physical layer in VANETs is based on the Dedicated short range communication (DSRC) protocol [14]. DSRC is a 75 MHz band in the 5.9 GHz frequency range with seven non-overlapping channels. Two different channel types are described for use in DSRC. The first type is the control channel, referred to as CCH, which is a single channel reserved for short, high-priority application and system control messages [13]. During the CCH, every node broadcasts a beacon that provides trajectory and other information about the vehicle. The other type of channel is the service channel, or SCH, which has six different 10 MHz channels that support a wider range of applications and data transfer. During CCH time channel activities on SCH are suspended and vice versa. BECSI uses the CCH to transmit the corresponding U_i. Each node in the VANET monitors the CCH during time periods designated as control channel intervals. The time period for an entire CCH Interval and SCH Interval is called a Sync Interval (see Fig. 6). Between CCH intervals, nodes may switch to participate on a SCH for applications such as file downloads.

Figure 6: CCH/SCH timing.

Each regional CA sends to RSUs an authenticated message M containing the corresponding U_i and a time-stamp.

$$CA \rightarrow RSUs : M = [U_i, TimeStamp]_{Sign_{CA}}$$

Note that regional CAs are expected to have a wireline to communicate with their corresponding RSUs. The time stamp included in the message allows vehicles to verify the freshness of the message. Thus, it is avoided potential forgery or replay attacks. The size of this message is 72 bytes:

- 64 bytes for the ECDSA-256 CA's signature.
- 4 bytes for the timestamp representing seconds UTC since the epoch ('1970-01-01 00:00:00' UTC).
- 4 bytes for representing the U_i value.

During the CCH interval, RSUs broadcast this message to OBUs in range. However, not in every CCH interval M is sent. Depending on the certificate revocation rate, each regional CA will choose the rate at which they have to transmit the U_i to the vehicles. Normally, vehicles will remain under the coverage of an RSU for more than $100ms$. Therefore, CAs have to adjust the frequency at which M is sent to avoid vehicles receiving multiple copies of the same message. Notice that as M is signed by the CA, any vehicle can act as repository and transmit this message without being able to modify it. The hash chain is initialized with secret nonce that the CA generates (U_d) and includes in the *extended*-CRL. By hashing U_d, the other U_i nodes of the chain are calculated. As the validity interval of the CRL is finite, the length of the chain is also finite, i.e., U_0 is the last node of the chain that is calculated after hashing d times U_d. Thus each value can be calculated applying a hash function h to the previous value, and the first value of the hash chain is the secret nonce U_d.

$$U_d \xrightarrow{h} U_{d-1} \xrightarrow{h} \ldots \xrightarrow{h} U_i \xrightarrow{h} U_2 \xrightarrow{h} U_1 \xrightarrow{h} U_0$$

On the other hand, BECSI not only issues delta-CRLs aperiodically, it also includes an extension in the delta-CRLs as it does with the base-CRL. Thus, any VANET entity that has cached the extended-delta-CRL can construct the BECSI Δ-tree (as shown in Sec.2.2). With this MHT, a non-TTP can respond to any entity requesting the status of a particular certificate. The response can be authenticated by the requesting party by means of the extension.

3.4.4 Certificate status checking

After the third phase, RSUs and some vehicles will be able to act as repositories. The last stage of the mechanism consists in providing the certificate status information to any vehicle that needs to validate the status of a certificate. Under BECSI, vehicles have to option to check the status of a certificate:

- Downloading the standard CRL and the available delta-CRLs from any repository. This option is desirable when the connectivity to the infrastructure is high and the network congestion is low. For instance, in a urban scenario during non-rush hours where the deployed infrastructure should be enough to serve the CSI.

- Requesting the status of a particular certificate to any repository. With this option, the requesting vehicle only gets the status of a single certificate, so the bandwidth load is low. Vehicles should use this option when they need a quick response (e.g. authenticating safety messages), or when the VANET conditions are not good enough to download the whole CRL and delta-CRLs.

Independently of the option, vehicles that need to check the status of a certificate must locate a valid repository. To do so, vehicles use a Service Discovery Protocol (SDP) to find a RSU or a vehicle that is acting as repository. Once the repository has been located, the vehicles can query for the CRL or query for the status of a particular certificate using the following status checking protocol.

The protocol for status information exchange is based on the hash tree structure and it allows checking the integrity of a single *extended-CRL* or *extended-delta-CRL* entry with only some hash material plus the $\mathcal{D}$igest (included in the extension) and the corresponding U_i. On the one hand, this is much more efficient than broadcasting the entire *extended-CRL* and the *extended-delta-CRLs*. On the other hand, the mechanism is fully offline (the only trusted authority is the CA), which is a very good feature because sometimes it may be impossible for vehicles to reach the CA due to lack of coverage.

Hence, a vehicle that needs to check the status of a certificate must follow the next steps:

1. The vehicle uses a service discovery protocol to find either a RSU or a mobile repository inside its coverage range for status checking.

2. The vehicle sends the serial number of the certificate that is going to be verified to the repository. The repository searches the target certificate

in the base-tree and the Δ-trees. In the case the certificate is found, the repository sends the $\mathcal{P}$ath, i.e., the hash values of the nodes of the base-tree (or Δ-tree) which are needed to calculate the signed root. To calculate the path, the repository follows a recursive algorithm that starts from the root and goes across the MHT until the target leaf is reached (see Algorithm 1).

Algorithm 1: Algorithm to calculate the $\mathcal{P}$ath of a given certificate.

```
Input: SN_target
Output: Path
foreach base-tree and Δ-tree_i do
    if SN_target ∈ Δ-tree_i then
        k = i
    else
        k = 0
N_ij = root_k;
while N_ij.max ≠ N_ij.min do
    i = i − 1
    j = 2 · j
    if N_ij.max < SN_target then
        Path.add(N_ij)
        j = j + 1
    else
        Path.add(N_{i,j+1})
return Path,k
```

3. The vehicle verifies that the H_{root} (or the H^i_{root}) calculated from the $\mathcal{P}$ath matches the H_{root} (or the H^i_{root}) contained in the $\mathcal{D}$igest$_{base}$ (or the $\mathcal{D}$igest$_{\Delta_i}$).

Notice that as all H_{root} and all $\mathcal{D}$igest are signed by the CA, it is just as impractical to create falsified values of the $\mathcal{P}$ath as it is to break a strong hash function. In case the certificate is not revoked, the repository sends the adjacent leaves to the requested certificate. To this respect, the repository has to prove that a certain certificate (SN_{target}) does not belong to the set of revoked certificates (Φ). To prove that $SN_{target} \notin \Phi$, as the leaves are ordered, it is enough to demonstrate the existence of two leaves, a minor adjacent (SN_{minor}) and a major adjacent (SN_{major}) for the base-tree and each Δ-tree that fulfill:

1. $SN_{major} \in \Phi$.

2. $SN_{minor} \in \Phi$.

3. $SN_{minor} < SN_{target} < SN_{major}$.

4. SN_{minor} and SN_{major} are adjacent nodes.

So in the worst case, where d delta-CRLs have been published, the repository will have to send $2d+1$ $\mathcal{P}$aths to proof that a certificate is not revoked, i.e, the serial number is not contained neither in the base-CRL nor in the delta-CRLs. Note that in any case, the amount of data necessary to proof that is smaller than the whole CRL. Therefore, checking vehicles have to know exactly the number of published Δ-trees, i.e., to check that a certificate is not revoked it must corroborate that it does not belong to any MHT. In any case, the data that the repository needs to send to a node to perform the status checking can be placed in a single UDP datagram using 802.11p link-layer.

4 Performance Evaluation

In this section, we evaluate the efficiency of the proposed status checking protocol and we compare it with other certificates status management protocols designed for VANET. First, we define a set of metrics to compare the performance of revocation schemes. Then, BECSI is evaluated through simulation using NCTUns [31]. NCTUns was chosen for its advanced IEEE 802.11 model library and ability to integrate with any Linux networking tools.

4.1 Comparison Criteria

- *Query Cost (Q_{cost})*: This criterion measures the cost of certificate validity checking. The cost represents bandwidth requirement from repositories to vehicles. Therefore, we calculate this cost as the size of a CSI query (s_q) plus the size of its response (s_r):

$$Q_{cost} = s_q + s_r. \tag{4}$$

- *Request Ratio*: This metric captures the amount of requests that the VANET entities perform to update the CSI. If client validation requests arrive independent of each other, an exponential inter-arrival probability density function can be used to derive the request rate (R) for downloaded CRLs as in [6]:

$$R_t = N_{veh}\lambda e^{\lambda t}, \tag{5}$$

where N_{veh} is the total number of vehicles in the VANET and λ is the ratio of certificates validated per day by each vehicle.

- *Window of vulnerability (WOV)*: This criterion captures the risk of operating with cached CSI. It indicates how long the new revocation data might be held by CAs before being distributed to vehicles. In this paper, WOV is measured in number of hours, which is reasonable because typically CRLs are normally updated every day. We estimate the WOV not only taking into account the validity interval of issued CSI but also the ratio of unknown revoked certificates during this interval as in [20]. Thus,

$$WOV(t) = \frac{\rho(t - t_0)}{(1-\rho)T_c + \rho(t - t_0)}, \tag{6}$$

where T_c is the mean certificate lifetime, ρ is the revocation ratio of revoked certificates, and t_0 is the issuing instant of the CSI.

- *Scalability*: This criterion shows how a revocation mechanism scales in large VANETs, measured as the ratio of increased costs (in terms of update and query costs) over increased size of the vehicles (measured in the number of certificates, queries and revoked certificates). If we assume a stable certificate revocation rate and query rate, a larger VANET typically indicates more revoked certificates and queries in unit time.

4.2 Analytical Evaluation

In this section, we compare analytically the performance of BECSI to other certificate status validation mechanisms. To that end, we compare BECSI with these mechanisms in terms of aforementioned metrics.

4.2.1 Query Cost Analysis

First of all, we start estimating the size of a CRL in a vehicular environment.

Fig. 7 describes the size of each of the elements that compose a CRL. Note that, in a VANET, the size of the CRL will depend mainly on the number of revoked certificates, so that the size of the CRL header is negligible compared to the total size of the CRL. Let N_{veh} be the total number of vehicles in the region that the CRL needs to cover, ρ the average percentage of certificates revoked,

CRL Header ($\sim$ 50 bytes)

- Issuer's name: 32 bytes (if X.500 name used)
- CRL issuance time (thisUpdate): 6 bytes
- Next CRL issuance time (nextUpdate): 6 bytes

List of revoked certificates (9 bytes per revoked certificate)

- Serial number : 3 bytes
- Revocation date : 6 bytes
- CRL entry extensions (e.g. revocation reason)

CRL general extensions (e.g. CRL Number)

Signature of CRL issuer (64 bytes for ECDSA-256 bit)

Figure 7: Key elements of X.509 v2 CRL

L_f the lifetime of a certificate, and $\bar{s}$ the mean number of pseudonyms of a vehicle. Additionally, let N_{rev} be the number of non-expired certificates that were revoked, i.e., the number of certificates that the CRL contains. According to [30], the probability of a certificate being revoked follows an exponential distribution. Then, the probability of a given certificate to become revoked at any time period of its lifetime $i \in [0, \ldots, L_f]$ can be expressed as:

$$P_{rev}(i) = L_f e^{-i \cdot L_f}.$$

When a certificate is revoked at time period i of its lifetime, it stays in the CRL for $L_f - i$ time periods. Thus, the expected time a revoked certificate stays in the CRL can be estimated as:

$$E(L_f - i) = E(L_f) - E(i) = L_f - \frac{1}{L_f} = \frac{L_f^2 - 1}{L_f} \simeq L_f.$$

Then, we can estimate the mean number of revoked certificates in a CRL as:

$$\overline{N_{rev}} = N_{veh} \cdot \rho \cdot \bar{s} \cdot L_f.$$

Finally, we estimate the size of a CRL in a VANET. As shown in Fig.. 7, CRL entries will have varying sizes, but according to 1609.2 standard [13], 14 bytes

per entry is a realistic figure, i.e, $s_e = 14$ bytes. The size of the CRL header is negligible compared to the total size of the CRL. According to NIST statistics [4], 10% of the certificates need to be revoked during a year, i.e., $\rho = 0.1$. Recall that in a VANET, each vehicle owes not only an identity certificate, but also several pseudonyms. The number of pseudonyms may vary depending on the degree of privacy and anonymity that it must be guaranteed. According to Raya, Papadimitratos, and Hubaux in [26] the OBU must store enough pseudonyms to change pseudonyms about every minute while driving. This equates to about 43,800 pseudonyms per year for an average of two hours of driving per day. Haas, Hu, and Laberteaux in [11] recommend changing pseudonyms every 10 minutes, and driving 15 hours per week. This equates to 4,660 pseudonyms per year, but they recommend storing five years of pseudonyms for a total of about 25,000 pseudonyms per OBU. Therefore, we set $\bar{s} = 25,000$. Regarding to the certificate lifetime, according to [30], it ranges from 26 to 37 days. In this manner, we set the lifetime to 1 month. Therefore, the expected CRL size is:

$$CRL_{size} = \overline{N_{rev}} \cdot s_e = N_{veh} \cdot \rho \cdot \bar{s} \cdot L_f \cdot s_e$$

Assuming that a regional certification authority manages a very short population of around 50,000 vehicles, the expected CRL size is $CRL_{size} \simeq 145$ Mbytes.

On the other hand, the response size of BECSI (when using the MHTs to generate authenticated responses) is much smaller than a CRL as it consists only of the $\mathcal{D}$igest and the $\mathcal{P}$ath for a given certificate. Using the SHA-1 algorithm (hash size of 160 bits), and ECDSA-256 the size of the response of BECSI for 10,000,000 revoked certificates (including pseudonyms) is of approximately 725 bytes.

Mechanism	**Request size**	**Response size**	Query Cost
CRL	73 bytes	145 Mbytes	∼145 Mbytes
Compressed CRL (Bloom Filter-2% false positives)	73 bytes	10 Mbytes	∼10 Mbytes
ADOPT	66 bytes	586 bytes	652 bytes
BECSI tree	73 bytes	725 bytes	778 bytes

Table 1: Comparison of the overhead introduced by BECSI and other certificate validation mechanisms.

In terms of the total overhead introduced to the network, Table 1 shows the *Query Cost* for current proposed certificate validation mechanisms. Note that

the request size is very similar for all the mechanisms. However, the size of the response varies significantly, e.g., BECSI and ADOPT response sizes are six orders of magnitude smaller than conventional CRL. Fig. 8 shows the size of the response for CRL, Compressed CRL, ADOPT [17] and BECSI depending on the number of revoked vehicles in the network. While ADOPT response size is constant, the size of the response when using CRL or a compressed version of the CRL increments with the number of revoked certificates. Notice that, the CRL size grows linearly with the number of revoked certificates, while BECSI response sizes describe a logarithmic growth. Therefore, in terms of Query Cost, BECSI is more efficient than CRL and the compressed CRL. Regarding ADOPT, its response size is slightly smaller than in BECSI, but it lacks of the benefits that BECSI provides to operate during disconnections. ADOPT relies on the fact that any vehicle stores the previously received CSI responses. In vehicular scenarios the number of cached responses could be huge, and therefore, also a huge storage capacity is required in the vehicle. In addition, ADOPT does not guarantee that a vehicle obtains the status of a given certificate when needed. So, ADOPT has smaller responses, but it does not provide as fresh information as BECSI, it forces VANET nodes to store a large amount of CSI data and finally, it makes the network more vulnerable due to the potential unavailability of required CSI.

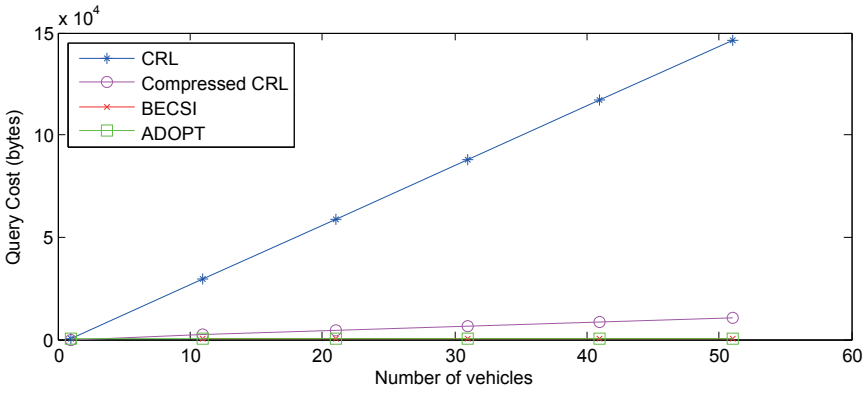


Figure 8: Response size vs. number of vehicles.

4.2.2 Request Ratio Analysis

In the traditional method of certificate revocation, each CRL includes a *nextUpdate* field that specifies the time at which the next CRL will be issued. Thus,

once a relying party has obtained a CRL in order to perform a validation, it will not need to request any further information from the repository to perform future validations until the time specified in the *nextUpdate* field of the CRL in its cache has been reached. So, during the period of time in which a CRL is valid (i.e., the most current), each relying party will make at most one request to the repository for revocation information. This request will be made the first time after the current CRL is issued that the relying party performs a validation. Thus, the request ratio of the CRL decreases during the validity interval of the CRL following an exponential function (see. Fig. 9). Figure 9 shows the request rate for a CRL, issued using the traditional method, over the course of 24 hours. The graph in this figure was drawn assuming that a CRL was issued at time 0 and that no other CRLs were issued during the period of time shown in the graph. It was also assumed that there are 50,000 vehicles each validating an average of 10 certificates per day.

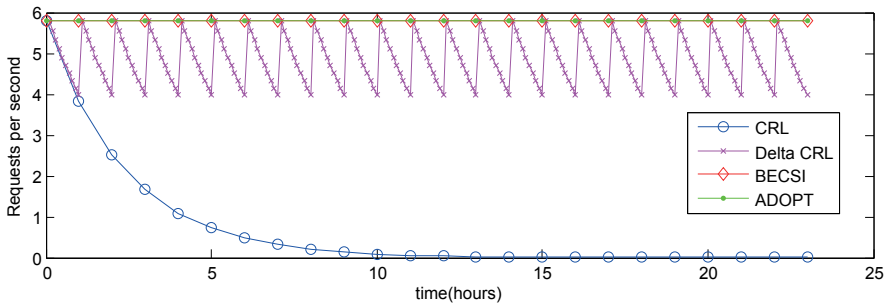


Figure 9: Request rate for different revocation mechanisms.

Figure 9 shows also an example of delta-CRLs issued in the traditional manner. In this example, vehicles download base CRLs at most once every 24 hours. Delta-CRLs are then obtained to ensure that validations are based on certificate status information that is at most 1 hour old. Each validation will require access to a delta-CRL and its corresponding base CRL (either downloaded from the repository or generated locally from a delta-CRL and a previous base CRL). So, the request rate for delta-CRLs will be the same as the request rate for full CRLs in a system that does not use delta-CRLs. Base CRLs, on the other hand, will be downloaded less frequently.

Finally, regarding the cases of BECSI and ADOPT, the request rate is almost constant, i.e., every time a vehicle needs to check the status of a particular certificate they must query a repository. Note that this rate decreases with

time, as vehicles also have the ability to store previously queried CSI. However, as the number of valid certificates is so large in VANETs, this decrement is imperceptible.

4.2.3 WOV Analysis

The window of vulnerability (WOV) affects update and query bandwidth requirements and/or repositories processing loads directly, while these two factors are two major features determining scalability of CSI issuing mechanisms. WOV presents a direct tradeoff between the security/ timeliness and system scalability. No window of vulnerability means high security and is thus desirable; however, it requires either timely certificate status update from CAs that can force a high update cost and incur security risk.

The traditional way of issuing CRL is the worst mechanism in terms of WOV. During the whole validity of the CRL, vehicles are unaware of new revoked certificates. Therefore, the WOV will increase during the validity of the CRL as there will be more unknown revoked certificates as times goes by. Figure 10 shows the WOV for a CRL issued periodically each 24 hours, and with a constant revocation rate $\rho = 0.1$. Note that the revocation rate determines the slope of the function, i.e., higher revocation rate will give higher WOV.

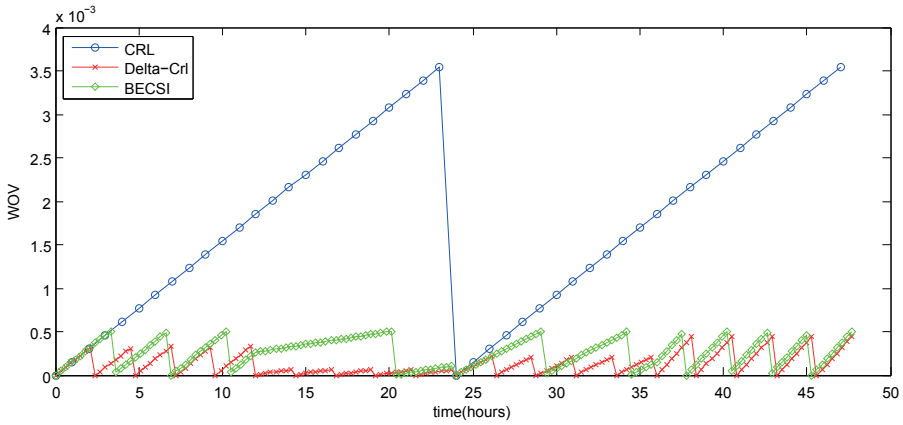


Figure 10: WOV for different revocation mechanisms.

Compressed CRLs have the same WOV that the standard CRL as they are just a compressed version of the CRL issued with the same lifetime. In the same way, ADOPT also has the same WOV that a CRL. ADOPT presents a distributed mechanism that takes advantages of V2V communications to issue

cached CSI. However, the CSI source of this cached revocation information is a CRL. Therefore, the validity period of the cached information in ADOPT is the same that the validity period of the source CRL, i.e., the WOV is the same.

Figure 10 also shows the WOV for BECSI and delta-CRLs. Note that both mechanisms improve the WOV. Traditional delta-CRLs reduce the WOV as they are issued periodically during the validity of the base-CRL. Therefore, the interval of vulnerability of the base-CRL is reduced as many times as delta-CRLs are issued during the lifetime of the base-CRL. In the example shown in the figure, for each base-CRL issued each day, a delta-CRL is issued each 2.4 hours. Thus, 10 delta-CRLs are issued during the validity interval of the base-CRL, reducing the WOV ten times.

In the same way, BECSI also uses delta-CRLs to construct the tree structure. Therefore, the tree structures used in such scheme reduces not only update or query costs, but also the WOV. Recall that BECSI does not issue delta-CRLs periodically, but these are issued with a fixed size. In this sense, despite that fact that the WOV could be higher than with the traditional delta-CRL issuing mechanism, the maximum WOV is always constant. Thus CAs can manage the WOV by selecting the size of the delta-CRLs. In the example shown in the figure 10, the number of delta-CRLs issued during the validity interval of the base-CRL is reduced compared to the traditional delta-CRL issuing mechanisms. Note that in this example, BECSI's WOV is never higher than 0.0005.

4.2.4 Scalability Analysis

When the vehicular population is large, CRLs tend to become large imposing high bandwidth costs on the CRL distribution points. Hence traditional CRL-based schemes do not scale well. If clients have limited bandwidth capability as is the case of the 802.11p, downloading large CRLs will be user-unfriendly.

With traditional delta-CRLs, the base-CRLs are issued less frequently (as shown in Fig. 5), this reducing the total bandwidth load on the CRL distribution points. However, that use of the traditional delta-CRL does not lead to a significant reduction in bandwidth as one would expect. If delta-CRLs are issued very frequently, there is no advantage in using traditional delta-CRLs. Therefore, although the scalability improves compared to simple CRLs mechanisms, traditional delta-CRLs scalability depends on the issuing periodicity of the delta-CRLs.

BECSI takes advantage of the delta-CRLs and optimize the issuing interval so that delta-CRLs remain constant in size. With BECSI, delta-CRLs always

have the same size, but they are issued aperiodically. Thus, BECSI becomes more scalable than traditional delta-CRL where depending on the revocation rate the issuing period of the delta-CRLs could be bandwidth-inefficient. Moreover, BECSI also takes advantage of the capabilities of the V2V communication, allowing any vehicle in the network to become a mobile repository. In this sense, BECSI (as ADOPT), multiplies the number of potential repositories, and, therefore, its scalability is also increased.

4.3 Simulation

In the previous section, we have seen analytically that BECSI mechanisms outperform CRL in terms of Query Cost, WOV and scalability. Moreover, BECSI also improves other revocation mechanisms such as ADOPT when analyzing the availability of fresh CSI. In this section, we evaluate the proposed mechanism in a VANET scenario taking into account the specific characteristics of these networks. Using the simulator NCTUns [31], BECSI is evaluated.

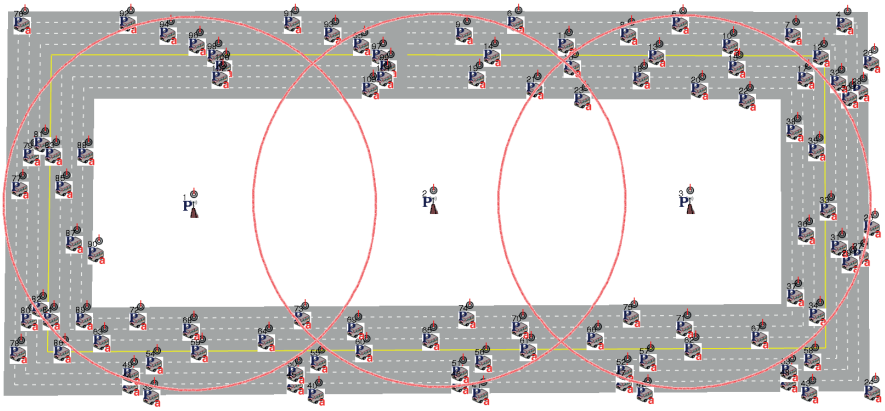


Figure 11: Simulation scenario.

The reference scenario is shown in Figure 11. This scenario consists of 4 two-lane roads forming a 1000x500m rectangle. Three RSUs are placed every 300 meters. Note that there are some areas of the highway that are not covered.

Table 2 summarizes the values of the configuration parameters used in the reference scenario. Note that we have configured our simulation to use the Nakagami propagation model. We choose this propagation model because empirical research studies have shown that a fading radio propagation model, such as the Nakagami model, is best for simulation of a vehicular environment [29].

Parameter	Value
Area	1000x500m
Number of RSUs	3
Number of OBUs	100
RSU Transmission range	300m
MAC	IEEE 802.11p
Propagation model	Nakagami
Number of caching nodes	20
Maximum speed	120 km/h

Table 2: Parameter values for the reference scenario.

Using this scenario as reference, firstly, we compare the CSI validation delay of the BECSI scheme with that of the classical PKI [13] under a well-deployed VANET. In the conducted simulation, we consider the cryptography delay only due to hashing operations and point multiplication operations on an elliptic curve, as they are the most time-consuming operations in the proposed protocol. Let T_{hash} and T_{mul} denote the time required to perform a pairing operation and a point multiplication, respectively, respectively. Elliptic curve digital signature algorithm is the digital signature method chosen by the VANET standard IEEE1609.2, where a certificate and signature verification takes $4T_{mul}$, and a signature generation takes T_{mul}. To verify a credential in the basic scheme described in Section 3.4.4, a vehicles must perform a hash operation to compute the current contents of leaf node in the BECSI-tree corresponding to SN_i. Finally, it performs $logN$ hash operations to compute the root of the BECSI-tree using the $\mathcal{P}$ath. Therefore, the total computation overhead when checking the status of a certificate is $T_{hash}(logN + 1) + 4T_{mul}$. In [36], T_{mul} are found for an MNT curve with embedding degree $k = 6$ that is equal to 0.6 ms. In our simulation, we use an Intel Core i7 950 (at 3.07GHz) which is able to perform 1015952 SHA-1 Hashes per second, i.e, $T_{hash} = 0,98\mu$s. Therfore the expected time to check the validity of a $\mathcal{P}$ath in BECSI with is 2.4 ms.

In VANETs, the most important issue in any revocation method is the delay of delivering the CSI to the vehicles to prevent that misbehaving vehicles from jeopardizing the safety of its neighbors. Consequently, we measure the revocation delay as delay from the moment a vehicle issues a CSI request until the moment the new CSI is received. Table 3 shows the average time spent by a vehicle to retrieve CSI from a repository.

It is worth noting that the worst mechanisms in terms of delay are the traditional CRL and delta-CRL as requesting entities are downloading all the

Revocation Mechanism	Average Time	Standard Deviation
CRL (300 KB)	2,23 min	0,51 min
Compressed-CRL (20 KB)	7,01 sec	1,12 sec
Traditional Delta CRL (2.5-15 KB)	4,47 sec	2,12 sec
ADOPT (652 B)	705,06 ms	200,81 ms
BECSI Delta CRL (8 KB)	6,02 sec	0,05 sec
BECSI MHT (778 B-912 B)	483,02 ms	20,31 ms

Table 3: Time required to retrieve CSI.

available CSI. However, the delay of the conventional CRL compared with the proposed BECSI protocol decreases with the number of CSI requests. The variations in time to download the CRL are due to the number of intermediate RSUs existing in the connection between the CA and the vehicle sending the revocation request. The average time to validate the status of a certificate in ADOPT is lower than BECSI because of the number of hops that are necessary to retrieve the cached CSI. BECSI in its MHT mode of operation is the fastest in average when validating the status of a certificate. However, this mode of operation has a also a notable deviation. While in ADOPT the high deviation is due to the number of hops, in BECSI this deviation is mainly due to the number of Δ-trees that a vehicle has to check when a certificate is not revoked. Note also, that there are also some deviations from the theoretical expected results. This is due to several reasons such as the non-uniform distribution of the mobile repositories, the distance to the repositories or the congestion of the channel. Figure 12 shows the number of vehicles that are able to download the CSI in a particular range time depending on the revocation mechanisms. As expected, with BECSI and ADOPT almost all the 100 vehicles are able to download and process the CSI in less than 1,5 seconds. However, with Delta-CRLs and compressed-CRLs it takes from 4 to 8 seconds to retrieve the CSI.

Finally, we also evaluate the overhead introduced by BECSI. BECSI introduces overhead due to the transmission of the value of the hash chain in the control channel. To evaluate this in the CCH channel, we configure the RSUs to transmit this message every second. As expected, the vehicle is receiving messages from the RSU in range every 100 ms; and every second it receives the message M that involves an increase of the incoming throughput of 72 bytes. In this sense, the overhead introduced by the BECSI mechanism is 4% of the toal capacity of the CCH channel.

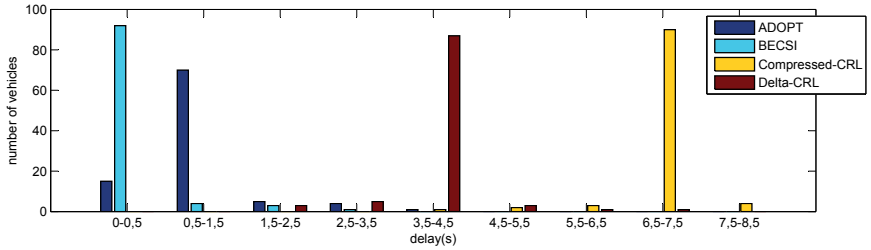


Figure 12: Histogram plot of time delay of the vehicles that receive the CSI depending on the revocation mechanism.

5 Conclusions

The revocation service is critical to permit efficient authentication in VANETs. Decentralized approaches based on reputation and voting schemes provide mechanisms for revocation management inside the VANET. However, the local validity of the CSI and the lack of support for extending its validity to the global VANET restrain their utilization in the real scenarios. The IEEE 1609.2 standard suggest the use of CRLs to manage the revocation data. However, the tradicional way of issuing CRLs do not fit well in a VANET where huge number of nodes are involved and where several pseudonym certificates are assigned in addition to vehicle identity certificates.

In this paper, we have presented BECSI, a bandwidth efficient certificate status checking mechanism based on the use of a hybrid delta-CRL scheme and MHTs. BECSI introduces an extension to both base-CRL and delta-CRL allowing any non-TTP to act as repository. The main advantage of this *extended-CRL* and *extended-delta-CRL* is that the road-side units and vehicles can build an efficient structure based on an authenticated hash tree to respond to status checking requests inside the VANET, saving time and bandwidth. Thus, vehicles do not have to download the whole CRL but query for the status of the certificate they need to operate with. Moreover, as *extended-delta-CRLs* have a fixed size, BECSI avoids the traditional problem of optimizing the validity windows of delta-CRLs. Thus, the risk of operating with unknown revoked certificates remains constant during the validity interval of the base-CRL, and CAs have the ability to manage this risk by setting the size of the delta-CRLs.

Analytical and simulation results show that allocating a small bandwidth is enough to ensure that vehicles receive CSI responses within few seconds. The performance improvement is obtained at expenses of adding the signed hash tree

extension to the standard-CRL. BECSI evaluation shows that not only improves in terms of bandwidth but also in terms of scalability (increase in the number of available repositories) and vulnerability (controlled WOV). In this way, BECSI becomes an offline certificate status validation mechanism as it does not need trusted responders to operate. Therefore, BECSI significantly achieves great efficiency and scalability, especially when deployed in heterogeneous vehicular networks.

Acknowledgments

This work is funded by the Spanish Ministry of Science and Education under the projects CONSOLIDER-ARES (CSD2007-00004) and TEC2011-26452 "SERVET", FPU grant AP2010-0244, and by the Government of Catalonia under grant 2009 SGR 1362.

References

[1] ITU-T X.509 : Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks, 2005.

[2] Frederik Armknecht, Andreas Festag, Dirk Westhoff, and Ke Zeng. Cross-layer privacy enhancement and non-repudiation in vehicular communication. In *4th Workshop on Mobile Ad-Hoc Networks (WMAN'07)*, 2007.

[3] R. Bera, J. Bera, S. Sil, S. Dogra, N. B. Sinha, and D. Mondal. Dedicated short range communications (DSRC) for intelligent transport system. In *Wireless and Optical Communications Networks, 2006 IFIP International Conference on*, pages 5 pp.+, 2006.

[4] S. Berkovits, S. Chokhani, J. Furlong, J.Geiter, and J. Guild. Public key infrastructure study: Final report. Technical report, MITRE Corporation for NIST, 1995.

[5] Igor Bilogrevic, Mohammadhossein Manshaei, Maxime Raya, and Jean-Pierre Hubaux. Optimal Revocations in Ephemeral Networks: A Game-Theoretic Framework. In *Proceedings of the 8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt 2010)*, pages 184–193. IEEE, 2010.

[6] D.A. Cooper. A model of certificate revocation. In *Fifteenth Annual Computer Security Applications Conference*, pages 256–264, 1999.

[7] Chun-I Fan, Ruei-Hau Hsu, and Chun-Hao Tseng. Pairing-based message authentication scheme with privacy protection in vehicular ad hoc networks. In *Proceedings of the International Conference on Mobile Technology, Applications, and Systems*, Mobility '08, pages 82:1–82:7, 2008.

[8] Jordi Forné, Jose L. Muñoz, Oscar Esparza, and Francisca Hinarejos. Certificate status validation in mobile ad hoc networks. *Wireless Commun.*, 16:55–62, February 2009.

[9] Carlos Gañán, Jose L. Muñoz, Oscar Esparza, Jorge Mata-Díaz, and Juanjo Alins. Toward revocation data handling efficiency in vanets. In *Proceedings of the 4th international conference on Communication Technologies for Vehicles*, Nets4Cars/Nets4Trains'12, pages 80–90, Berlin, Heidelberg, 2012. Springer-Verlag.

[10] C. Gañán, J. L. Muñoz, O. Esparza, J. Mata, J. Hernández-Serrano, and J. Alins. Coach: Collaborative certificate status checking mechanism for vanets. *Journal of Network and Computer Applications*, 2012.

[11] Jason J. Haas, Yih-Chun Hu, and Kenneth P. Laberteaux. Design and analysis of a lightweight certificate revocation mechanism for vanet. In *Proceedings of the sixth ACM international workshop on VehiculAr Inter-NETworking*, VANET '09, pages 89–98, New York, NY, USA, 2009. ACM.

[12] J.J. Haas, Yih-Chun Hu, and K.P. Laberteaux. Efficient certificate revocation list organization and distribution. *Selected Areas in Communications, IEEE Journal on*, 29(3):595 –604, march 2011.

[13] IEEE. IEEE trial-use standard for wireless access in vehicular environments - security services for applications and management messages. *IEEE Std 1609.2-2006*, pages 1–117, 2006.

[14] D. Jiang and L. Delgrossi. IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 2036–2040, May 2008.

[15] Kenneth P. Laberteaux, Jason J. Haas, and Yih-Chun Hu. Security certificate revocation list distribution for vanet. In *Proceedings of the fifth*

ACM international workshop on VehiculAr Inter-NETworking, VANET '08, pages 88–89, 2008.

[16] Leslie Lamport. Password authentication with insecure communication. *Commun. ACM*, 24:770–772, November 1981.

[17] G. F. Marias, K. Papapanagiotou, and P. Georgiadis. Adopt. a distributed ocsp for trust establishment in manets. *11th European Wireless Conference 2005*, 2005.

[18] R.C. Merkle. A certified digital signature. In *Advances in Cryptology (CRYPTO89). Lecture Notes in Computer Science*, number 435, pages 234–246. Springer-Verlag, 1989.

[19] Tyler Moore, Jolyon Clulow, Shishir Nagaraja, and Ross Anderson. New strategies for revocation in ad-hoc networks. In *Proceedings of the 4th European conference on Security and privacy in ad-hoc and sensor networks*, ESAS'07, pages 232–246, 2007.

[20] Jose L. Muñoz, Oscar Esparza, Carlos Gañán, and Javier Parra-Arnau. Pkix certificate status in hybrid manets. In *WISTP*, volume 5746 of *Lecture Notes in Computer Science*, pages 153–166. Springer, 2009.

[21] M.E. Nowatkowski, J.E. Wolfgang, C. McManus, and H.L. Owen. The effects of limited lifetime pseudonyms on certificate revocation list size in vanets. In *IEEE SoutheastCon 2010 (SoutheastCon), Proceedings of the*, pages 380 –383, march 2010.

[22] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Zhendong Ma, F. Kargl, A. Kung, and J.-P. Hubaux. Secure vehicular communication systems: design and architecture. *Communications Magazine, IEEE*, 46(11):100 –109, November 2008.

[23] P. Papadimitratos, L. Buttyan, J.-P. Hubaux, F. Kargl, A. Kung, and M. Raya. Architecture for secure and private vehicular communications. In *Telecommunications, 2007. ITST '07. 7th International Conference on ITS*, pages 1 –6, June 2007.

[24] Panagiotis Papadimitratos, Ghita Mezzour, and Jean-Pierre Hubaux. Certificate revocation list distribution in vehicular communication systems. In *Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking*, VANET '08, pages 86–87, 2008.

[25] Maxim Raya and Jean-Pierre Hubaux. The security of vehicular ad hoc networks. In *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, SASN '05, pages 11–21, 2005.

[26] Maxim Raya, Daniel Jungels, Panos Papadimitratos, Imad Aad, and Jean-Pierre Hubaux. Certificate revocation in vehicular networks. Technical report, EPFL, 2006.

[27] Maxim Raya, Mohammad Hossein Manshaei, Márk Félegyhazi, and Jean-Pierre Hubaux. Revocation games in ephemeral networks. In *Proceedings of the 15th ACM conference on Computer and communications security*, CCS '08, pages 199–210, 2008.

[28] S. Santesson and P. Hallam-Baker. Online Certificate Status Protocol Algorithm Agility. RFC 6277 (Proposed Standard), June 2011.

[29] Vikas Taliwal, Daniel Jiang, Heiko Mangold, Chi Chen, and Raja Sengupta. Empirical determination of channel characteristics for dsrc vehicle-to-vehicle communication. In *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*, VANET '04, pages 88–88, New York, NY, USA, 2004. ACM.

[30] Daryl Walleck, Yingjiu Li, and Shouhuai Xu. Empirical analysis of certificate revocation lists. In *Proceeedings of the 22nd annual IFIP WG 11.3 working conference on Data and Applications Security*, pages 159–174, 2008.

[31] S. Y. Wang and C. L. Chou. Nctuns tool for wireless vehicular communication network researches. *Simulation Practice and Theory*, 17:1211–1226, 2009.

[32] A. Wasef, Yixin Jiang, and Xuemin Shen. DCS: An Efficient Distributed-Certificate-Service Scheme for Vehicular Networks. *Vehicular Technology, IEEE Transactions on*, 59(2):533 –549, feb. 2010.

[33] A. Wasef and X. Shen. EMAP: Expedite Message Authentication Protocol for vehicular ad hoc networks. *Mobile Computing, IEEE Transactions on*, PP(99):1, 2011.

[34] A. Wasef and Xuemin Shen. EDR: Efficient Decentralized Revocation Protocol for Vehicular Ad Hoc Networks. *Vehicular Technology, IEEE Transactions on*, 58(9):5214 –5224, nov. 2009.

[35] A. Wasef and Xuemin Shen. Maac: Message authentication acceleration protocol for vehicular ad hoc networks. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1 –6, 30 2009-dec. 4 2009.

[36] Chenxi Zhang, Rongxing Lu, Xiaodong Lin, Pin-Han Ho, and Xuemin Shen. An efficient identity-based batch verification scheme for vehicular sensor networks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 246 –250, april 2008.

Toward Revocation Data Handling Efficiency in VANETs

Carlos Gañán, Jose L. Muñoz, Oscar Esparza
Jorge Mata-Díaz and Juanjo Alins

Universitat Politècnica de Catalunya (Departament Enginyeria Telemàtica)**
{carlos.ganan, jose.munoz, oesparza, jmata,juanjo}@entel.upc.es

Abstract. Vehicular Ad Hoc Networks (VANETs) require some mechanism to authenticate messages, identify valid vehicles, and remove misbehaving ones. A Public Key Infrastructure (PKI) can provide this functionality using digital certificates, but needs an efficient mechanism to revoked misbehaving/compromised vehicles. The IEEE 1609.2 standard states that VANETs will rely on the use of certificate revocation lists (CRLs) to achieve revocation. However, despite their simplicity, CRLs present two major disadvantages that are highlighted in a vehicular network: CRL size and CRL request implosion. In this paper, we point out the problems when using CRLs in this type of networks. To palliate these issues, we propose the use of Authenticated Data Structures (ADS) that allow distributing efficiently revocation data. By using ADS, network entities can check the status of a certificate decreasing the peak bandwidth load in the distribution points.

Keywords: Certification, PKI, Authenticated Data Structures.

1 Introduction

In the last decade, wireless communication between vehicles have drawn extensive attention for their promise to contribute to a safer, more efficient, and more comfortable driving experience in the foreseeable future. This type of communications have stimulated the emergence of Vehicular ad hoc networks (VANETs) which consist of mobile nodes capable of communicating with each other (i.e. Vehicle to Vehicle Communication -V2V communication) and with the static infrastructure (i.e. Vehicle to Infrastructure Communication -V2I communication). To make these communications feasible, vehicles are equipped with on-board units (OBUs) and fixed communication units (road-side units, RSUs) are placed

** This work is funded by the Spanish Ministry of Science and Education under the projects CONSOLIDER-ARES (CSD2007-00004) and TEC2011-26452 "SERVET", and by the Government of Catalonia under grant 2009 SGR 1362.

along the road. Applying short range wireless technology based on IEEE 802.11, multi-hop communication facilitates information exchange among network nodes that are not in direct communication range [1].

However, the open-medium nature of these networks and the high-speed mobility of a large number of vehicles make necessary the integration of primary security requirements such as authentication, message integrity, non-repudiation, and privacy [2]. Without security, all users would be potentially vulnerable to the misbehavior of the services provided by the VANET. Hence, it is necessary to evict compromised, defective, and illegitimate nodes. The basic solution envisioned to achieve these requirements is to use digital certificates linked to a user by a trusted third party. These certificates can then be used to sign information. Most of the existing solutions manage these certificates by means of a central Certification Authority (CA) [3]. According to IEEE 1609.2 standard [4], vehicular networks will rely on the public key infrastructure (PKI). In PKI, a CA issues an authentic digital certificate for each node in the network. Therefore, an efficient certificate management is crucial for the robust and reliable operation of any PKI. A critical part of any certificate-management scheme is the revocation of certificates.

Regarding the revocation of these certificates, some proposals allow revocation without the intervention of the infrastructure at the expense of trusting other vehicles criteria; and other proposals are based on the existence of a central entity, such as the CA, which is in charge of taking the revocation decision for a certain vehicle. Again, according to the IEEE 1609.2 standard [4], vehicular networks will rely on the existence of a CA. In this sense, it is stated that these networks will depend on certificate revocation lists (CRLs) and short-lived certificates to achieve revocation. CRLs can be seen as black lists that enumerate revoked certificates along with the date of revocation and, optionally, the reasons for revocation.

As the network scale of VANETs is expected to be very large and to protect the privacy of users each vehicle has many temporary certificates (or called pseudonyms), the CRLs are expected to be quite large. Moreover, CRLs have also associated a problem of request implosion, i.e., vehicles may become synchronized around CRL publication instant, as they may request CRL at or near the moment of publication. This burst of requests may cause network congestion that may introduce longer latency in the process of validating a certificate. To reduce the potential network and computational overhead imposed by any CRL distribution mechanism, some optimizations for organizing, storing, and exchanging CRL information have been proposed. In [2, 5], it is proposed a way of

compress CRLs using Bloom filters. Their method reduces the size of a CRL by using about half the number of bytes to specify the certificate serial number for revocation. However, the use of this probabilistic structure has associated a false positive rate that diminishes the efficiency of the revocation service.

In this paper, we explore the benefits of using authenticated data structures (ADS), such as binary trees or skip lists, to manage revocation data in VANETS. These structures are a model of computation where untrusted responders answer certificate status queries on behalf of the CA and provide a proof of the validity of the answer to the user. Although VANETs can greatly benefit from the use of ADSs, to the best of our knowledge there has been no proposal of deploying the revocation service by means of an ADS. By using these structures, both CRL issues are palliated: the CA is no longer a bottleneck as there are several responders that act on its behalf; and the revocation data can be checked without downloading the whole CRL.

2 CRLs' problematic in VANETs

As stated in the trial-use standard [4], for a certificate authority (CA) to invalidate a vehicle's certificates, the CA includes the certificate serial number in the CRL. The CA then distributes the CRL so that vehicles can identify and distrust the newly revoked vehicle. The distribution should spread quickly to every vehicle in the system.

However, the distribution itself poses a great challenge due to the size of the CRL. As a CRL is a list containing the serial numbers of all certificates issued by a given certification authority (CA) that have been revoked and have not yet expired, its distribution causes network overhead. Moreover, the CRL size increases dramatically if only a small portion of the OBUs in the VANET is revoked. To have an idea of how big the CRL size can be, consider the case where 1% of the total number of the OBUs in the United States is revoked. Recall that in a VANET, each vehicle owes not only an identity certificate, but also several pseudonyms. The number of pseudonyms may vary depending on the degree of privacy and anonymity that it must be guaranteed. According to Raya, Papadimitratos, and Hubaux in [5] the OBU must store enough pseudonyms to change pseudonyms about every minute while driving. This equates to about 43,800 pseudonyms per year for an average of two hours of driving per day. In the United States alone, 255,917,664 "highway" registered vehicles were counted in 2008, of which 137,079,843 passenger cars [6]. In

this case, the CRL would contain around 100 billion revoked certificates. Assuming that certificates can be identified by a 16 byte fingerprint (the size of one AES block), the CRL size would be of 1,7 TB approximately. Only the amount of memory necessary to storage this CRL makes it impossible its deployment. Therefore, the CRL size has to be reduced.

The CRL size can be reduced by using regional CAs. However, there appears a trade-off between the size of the CA region and size of the CRL, as well as the management complexity of the entire PKI system for VANETs. The least complicated region to manage would be a single large area, such as the entire United States, with a single CA responsible for every certificate and pseudonym. However, this gives place to CRLs of several terabytes. Therefore, it is necessary to divide the CRL information according to regional areas. In this sense, if we divide the entire United States by cities (i.e. 10,016 cities according to the U.S. census bureau), the CRL size is reduced to around 170 Mbytes. Using the 802.11a protocol to communicate with RSUs in range, vehicles could have between 10-30 Mbps depending on the vehicle's speed and the road congestion. Therefore, in the best case a vehicle will need more than 45 seconds to download the whole CRL. Under non-congested conditions, any vehicle should be able to contact the infrastructure for more than 45 seconds, and therefore download the CRL. In scenarios where vehicles are not able to keep a permanent link with the infrastructure for this amount of time, techniques such as Bloom filter or Digital Fountain Codes could be used to download the CRL. Therefore, though the problem of having a huge CRL is mitigated by the use of such techniques, the restraints imposed by the distribution affect the freshness of the revocation data.

A direct consequence of this significant time to download a CRL is that a new CRL cannot be issued very often, so its validity period has to be shortened. This validity period directly determines how often a vehicle has to update the revocation information. Therefore, the validity period of the CRL is critical to the bandwidth consumption. In this context, it appears another trade-off between the freshness of revocation information and the bandwidth consumed by downloading CRLs. Large validity periods will decrease the network overhead at expenses of having outdated revocation information. Small validity periods will increase the network overhead but users will have fresh information about revoked certificates. As CRLs cannot be issued every time there is a new revoked certificate, vehicles will be operating with revocation information that is not comprehensive. Therefore, they will be taking certain risk of trusting a certificate that could be potentially revoked.

3 Using Authenticated Data Structures for certificate revocation in VANETs

By replicating revocation data at untrusted responders near users, VANETs can enhance its performance but that replication causes a major security challenge. Namely, how can a vehicle verify that the revocation data replicated at the RSUs are the same as the original from the CA? A simple mechanism to achieve the authentication of replicated revocation data consists of having the digitally sign each revocation entry and replicating the CA signature too. However, in VANETs where the revocation data evolves rapidly over time, this solution is inefficient. To achieve higher communication and computation efficiency, we propose the use of authenticated data structures (ADS) to handle the revocation service in VANETs. ADSs are a model of computation where untrusted responders answer certificate status queries on behalf of the CA and provide a proof of the validity of the answer to the user. In this section, first we introduce the architecture necessary to adopt ADSs. Then, we describe different ADSs and their main benefits.

3.1 System Architecture

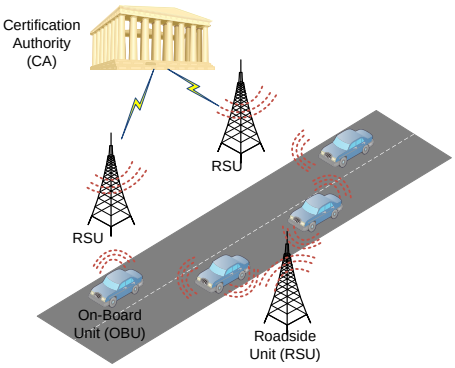


Fig. 1. System Architecture.

The system architecture to support ADSs consists in an adaptation of a PKI system to the vehicular environment. The ADS model involves a structured collection $\mathcal{R}$ of revoked certificates and three parties: the certification authority (CA), the road side units (RSUs), and the vehicles. A repertory of query operations and optional update operations are

assumed to be defined over $\mathcal{R}$. These three parties present a hierarchical architecture (see Fig. 1) which consists of three levels: the CA is located at level 1, as it is the top of the system. RSUs are located at level 2. Finally, the on-board units (OBUs) are located at the bottom of the hierarchy. Note that without loss of generality we consider a single the central trusted authority at the root, but it could be further divided into different state level trusted authorities and additionally a group of city level trusted authorities can be placed under every state authority.

The main tasks of each entity are:

1. The CA is responsible for generating the set of certificates that are stored in each OBU. It is also responsible for holding the original version of $\mathcal{R}$ and making it accessible to the rest of the entities. By definition of TTP, the CA should be considered fully trusted by all the network entities, so it should be assumed that it cannot be compromised by any attacker. In fact, in our proposal the CA is the only trusted entity within the network. Whenever an update is performed on $\mathcal{R}$, the CA produces structure authentication information, which consists of a signed time-stamped statement about the current version of $\mathcal{R}$.
2. RSUs are fixed entities that are fully controlled by the CA. They can access the CA anytime because they are located in the infrastructure-side, which does not suffer from disconnections. RSUs maintain a copy of $\mathcal{R}$. They interact with the CA by receiving from the CA the updates performed on $\mathcal{R}$ together with the associated structure authentication information. RSUs also interact with vehicles by answering queries on $\mathcal{R}$ posed by the vehicles. In addition to the answer to a query, RSUs also return answer authentication information, which consists of (i) the freshest structure authentication information issued by the CA; and (ii) a proof of the authenticity of the answer. If the CA considers that an RSU has been compromised, the CA can revoke it.
3. OBUs are in charge of storing all the certificates that a vehicle possesses. An OBU has abundant resources in computation and storage and allows any vehicle to communicate with the infrastructure and with any other vehicle in its neighborhood. OBUs pose queries on $\mathcal{R}$, but instead of contacting the CA directly, it contacts the RSU in range. However, OBUs only trust the CA and not the RSU about $\mathcal{R}$. Hence, it verifies the answer from the RSU using the associated answer authentication information.

3.2 System Requirements

- *Low computational cost*: The computations performed internally by each entity (CA, RSU, and OBU) should be simple and fast.
- *Low communication overhead*: CA-to-RSU communication (update authentication information) and RSU-to-OBU communication (answer authentication information) should be as small as possible.
- *High security*: the authenticity of the answers given by a RSU should be verifiable.

3.3 Authenticated Data Structures

Several ADSs have been proposed in the literature (mainly in the context of data base management) that fulfill the aforementioned requirements. In this section, we describe a repertoire of ADSs and to what extent they are capable of improving the revocation service.

Merkle Hash trees A Merkle hash tree (MHT) [7] is essentially a tree structure that is built with a collision-resistant hash function to produce a short cryptographic description of $\mathcal{R}$. The leaf nodes hold the hash values of the data of interest, i.e., the serial number of the revoked certificates (SN_1,SN_2,...,SN_n); and the internal nodes hold the hash values that result from applying the hash function to the concatenation of the hash values of its children nodes. In this way, a large number of separate data can be tied to a single hash value: the hash at the root node of the tree. MHTs can be used to provide an efficient and highly-scalable way to distribute revocation information.

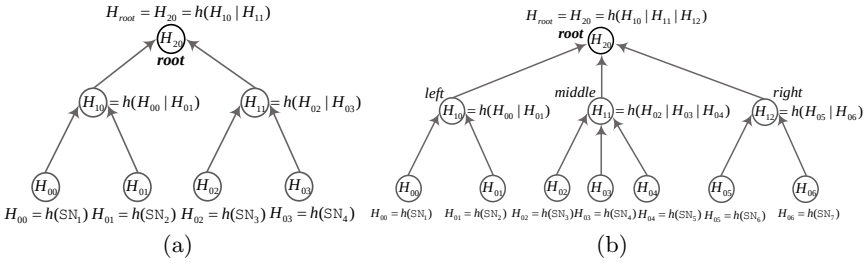


Fig. 2. Sample trees (a) MHT (b) 2-3.

A sample MHT is presented in Figure 2(a). The authentication of an element is performed using a verification path, which consists of the

sibling nodes of the nodes on the path from the leaf associated with the element to the root of the tree. The root value is signed and the collision-resistant property of the hash function is used to propagate authentication from the root to the leaves. This construction is simple and efficient and achieves signature amortization, where only one digital signature is used for signing a large collection of data. The hash tree uses linear space and has $O(log\ n)$ (where n denotes the number of revoked certificates) proof size, query time and verification time. An ADS based on hash trees can also achieve $O(log\ n)$ update time.

2-3 trees A standard 2-3 tree [8] is a tree where all leaves are at the same height and each node (except leaves) has two or three children. It has the nice property that leaf removal and insertion incur only logarithmic complexity because these operations only involve the nodes related to the path from the relevant leaf to the root.

Each leaf of such a 2-3 tree stores an element of set $\mathcal{R}$, and each internal node stores a one-way hash of its children's values. Thus, the CA-to-RSU communication is reduced to $O(1)$ entries, since the CA sends insert and remove instructions to the RSUs, together with a signed message consisting of a timestamp and the hash value of the root of the tree. RSUs respond to a membership query for an element SN_i as follows: if SN_i is in $\mathcal{R}$, then RSUs provide the path from the leaf storing SN_i to the root, together with all the siblings of the nodes on this path; else (SN_i is not in$\mathcal{R}$), RSUs provide the leaf-to-root paths from two consecutive leaves storing SN_j and SN_k such that $j < i < k$, together with all siblings of the nodes on these paths. By tracing these paths, OBUs can recompute the hash values of their nodes, ultimately recomputing the hash value for the root, which is then compared against the signed hash value of the root for authentication. As with MHTs, these trees achieve $O(log\ n)$ proof size, query time, update time and verification time.

One-way accumulator One-way accumulator (OWA) functions [9] allow a CA to digitally sign a collection of objects as opposed to a single one. The main advantage of this approach is that the validation of a response takes constant time and requires computations simple enough to be performed in resource-constrained devices. This type of ADS achieves a tradeoff between the cost of updates at the CA and queries at the RSUs, with updates taking $O(k + log(\frac{n}{k}))$ time and queries taking $O(\frac{n}{k})$ time, for any fixed integer parameter $1 \leq k \leq n$. For instance, one can achieve $O(\sqrt{n})$ time for both updates and queries.

Skip Lists Skip lists [10] are probabilistic ADSs that provide an alternative to balanced tree. Skip lists are sorted linked lists with extra links, designed to allow fast search in $\mathcal{R}$ by taking "shortcuts". The main idea is to enhance linked lists, which connect each element in the data sequence to its successor, by also connecting some elements to successors further down the sequence. Roughly half of the elements have links to their two-hop successor, roughly a quarter of the elements have links to their four-hop successor, and so on. As a result, during traversal from SN_i to element SN_j, the traversal path follows repeatedly the longest available link from the current element that does not overshoot the destination SN_j, and thereby reaches SN_j in fewer steps than would be possible by just traversing every intervening element between SN_i and SN_j. Compared with balanced trees, a skip list presents the following benefits:

- It is easy to implement and practically efficient in search, especially update time.
- It is space compact, where space is allocated when needed, while empty space is preserved in balanced tree.
- It is main memory index, while balanced tree are disk-based index.

Finally, Table 1 shows a comparison of the asymptotic performance of the main ADS versus traditional revocation mechanisms such as CRL or OCSP. Note that with ADSs, the revocation service can be greatly improved both in computation and communication overhead.

method	space	update time	update size	query time	query size	verifying time
CRL	$O(n)$	$O(1)$	$O(1)$	$O(n)$	$O(n)$	$O(n)$
OCSP	$O(n)$	$O(n)$	$O(n)$	$O(1)$	$O(1)$	$O(1)$
MHT	$O(n)$	$O(log\ n)$	$O(1)$	$O(log\ n)$	$O(log\ n)$	$O(log\ n)$
2-3 tree	$O(n)$	$O(log\ n)$	$O(1)$	$O(log\ n)$	$O(log\ n)$	$O(log\ n)$
Skip Lists	$O(n)$	$O(log\ n)$	$O(1)$	$O(log\ n)$	$O(log\ n)$	$O(log\ n)$
OWA	$O(n)$	$O(k + log(\frac{n}{k}))$	$O(k)$	$O(\frac{n}{k})$	$O(1)$	$O(1)$

Table 1. Comparison of the main ADS vs traditional revocation mechanisms.

3.4 Certificate Status Validation Protocol

The certificate status validation protocol consists in three stages.

1. *Revocation Service Setup*: The CA creates a CRL by appending the serial number of any revoked certificate. Then, it computes the corresponding ADS from the set $\mathcal{R}$ of revoked certificates contained in the

CRL. Once the ADS is computed, the CA signs the resulting time-stamped digest of the data structure, i.e., a collision resistant succinct representation of the data structure. The digest is transmitted to all the RSUs via a secure wireline together with the corresponding CRL. RSUs can either implement a push or pull protocol to transmit the digest to the vehicles in range.

2. *Certificate Status Updating*: Depending on the CA's policy, when an update is necessary, the CA recomputes the ADS and generates a new signed digest that is transmitted to the RSUs. Note that depending on the ADS, the data structure should be computed again or only update and delete operations should be performed. The new ADS is transmitted to the RSUs again, so that they could answer to validation queries.
3. *Certificate Status Querying*: OBUs query any RSU in range about the status of a particular certificate (SN_i). If $SN_i \in \mathcal{R}$, then the RSU computes the path necessary to allow OBUs to compute the digest and check that it matches the signed digest. If $SN_i \notin \mathcal{R}$, then the RSU computes the path of two consecutive certificates in $\mathcal{R}$ and transmit them to the requesting OBU. This OBU can then recompute the digest for both revoked certificates and be sure that $SN_i \notin \mathcal{R}$.

4 Evaluation

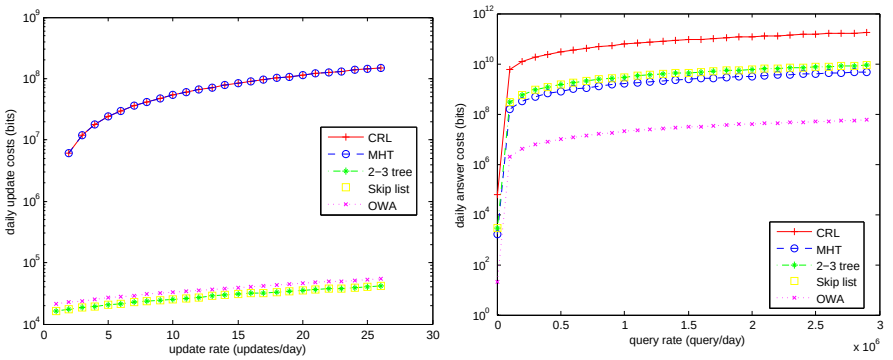
In the following, we compare the communication costs of using ADSs with the tradition CRL mechanism. To that end we define a set of parameters (see Table 2).

Parameter	Meaning of the parameter
N	Total number of certificates ($n = 3,000,000$)
k	Average number of certificates handled by a CA ($k = 30,000$)
p	Percentage of revoked certificates($p = 0.1$)
q	Number of certificate status queries issued per day ($q = 3,000,000$)
T	Number of updates per day ($T = 1$)
s_{SN}	Size of a serial number ($s_{SN} = 20$)
s_{sig}	Size of a signature ($s_{sig} = 1,000$)
s_{hash}	Size of the hash function ($s_{hash} = 128$).

Table 2. Notation

Using this notation, the CRL daily update cost is $T \cdot n \cdot p \cdot s_{SN}$ as each CA sends the whole CRL to the corresponding RSUs in each update.

The CRL daily query cost is $q \cdot p \cdot k \cdot s_{SN}$ as for every query the RSU sends the whole CRL to the querying OBU. When using ADS, these costs are drastically reduced. Note that no matter the type of ADS, OBUs do not have to download the whole CRL, and they only download status information about the certificate they want to operate with. Regarding MHTs, the RSUs have to recompute the tree in each update, so that daily update cost is $T \cdot n \cdot p \cdot s_{SN}$. However, to answer an OBU's query the RSU only needs to send up to $1 + log_2(pk)$ numbers, resulting in $q \cdot s_{hash}(1+log_2(pk))$ bits. In the case of 2-3 trees, to update the directory, the CA sends difference lists of total daily length of $\frac{n \cdot p \cdot s_{SN}}{365} + T \cdot s_{sig}$; and answer to OBUs' queries results in $2 \cdot q \cdot s_{hash} \cdot log_2(pk)$ bits. Similarly, skip lists need $2log_2\lceil pk \rceil$ number to answer an OBU's query and the same update cost than the 2-3 tree. With OWAs, the size of answer are drastically reduced to roughly s_{sig}, and the update cost depends on the accumulator configuration. We use Matlab R2011b to evaluate these costs.



(a) (b)

Fig. 3. (a) Daily CA-to-RSU update costs vs. update rate, (b) RSU-to-OBU query cost vs query rate.

Note that the costs will vary mainly depending on the total number of revoked certificates, the update rate and the number of queries. Figure 3(a) shows how the CA-to-RSU update communication costs of the different revocation mechanisms depend on the update rate (all other parameters are held constant). Note that any ADS is much more robust and efficient than CRL, even allowing once per hour updates. Regarding the query costs, as ADSs have smaller proof to validate the status of a

certificate they provide a more bandwidth efficient solution than CRL (see Fig. 3(b)).

5 Conclusions

In this paper, we consider the problem of certificate authentication and revocation in VANETs. We have proposed the use of authenticated data structures to handle the revocation service over VANETs. After discussing the issues of deploying CRLs in these environments, we show that ADSs are more robust to changes in parameters, and allow higher update/query rates than traditional revocation mechanisms. In addition, the adoption ADS reduces both the communication and the computational overhead in the OBUs. For our future work, we will investigate the use of mobile repositories under the context of the proposed schemes.

References

1. D. Jiang and L. Delgrossi. IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 2036–2040, May 2008.
2. Maxim Raya and Jean-Pierre Hubaux. The security of vehicular ad hoc networks. In *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, SASN '05, pages 11–21, 2005.
3. P. Papadimitratos, L. Buttyan, J.-P. Hubaux, F. Kargl, A. Kung, and M. Raya. Architecture for secure and private vehicular communications. In *Telecommunications, 2007. 7th International Conference on ITS*, pages 1 –6, June 2007.
4. IEEE trial-use standard for wireless access in vehicular environments - security services for applications and management messages. *IEEE Std 1609.2-2006*, pages 1–105, 2006.
5. Jason J. Haas, Yih-Chun Hu, and Kenneth P. Laberteaux. Design and analysis of a lightweight certificate revocation mechanism for vanet. In *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking*, VANET '09, pages 89–98, New York, NY, USA, 2009. ACM.
6. Bureau of Transportation Statistics U.S. Department of Transportation. Number of u.s. aircraft, vehicles, vessels, and other conveyances. `http://www.bts.gov/publications/national_transportation_statistics/html/table_01_11.html`, 2009. [Online; accessed 31-July-2011].
7. R.C. Merkle. A certified digital signature. In *Advances in Cryptology (CRYPTO89). Lecture Notes in Computer Science*, number 435, pages 234–246. Springer-Verlag, 1989.
8. M. Naor and K. Nissim. Certificate Revocation and Certificate Update. *IEEE Journal on Selected Areas in Communications*, 18(4):561–560, 2000.
9. Josh Benaloh and Michael de Mare. One-way accumulators: a decentralized alternative to digital signatures. In *Workshop on the theory of cryptographic techniques on Advances in cryptology*, EUROCRYPT '93, pages 274–285, 1994.
10. William Pugh. Skip lists: a probabilistic alternative to balanced trees. *Commun. ACM*, 33:668–676, June 1990.

Impact of the revocation service in PKI prices

Carlos Gañán, Jose L. Muñoz, Oscar Esparza
Jorge Mata-Díaz and Juanjo Alins

Universitat Politècnica de Catalunya (Departament Enginyeria Telemàtica)**
{carlos.ganan, jose.munoz, oesparza, jmata,juanjo}@entel.upc.es

Abstract. The ability to communicate securely is needed for many network applications. Public key infrastructure (PKI) is the most extended solution to verify and confirm the identity of each party involved in any secure transaction and transfer trust over the network. One of the hardest tasks of a certification infrastructure is to manage revocation. Research on this topic has focused on the trade-offs that different revocation mechanisms offer. However, less effort has been paid to understand the benefits of improving the revocation policies. In this paper, we analyze the behavior of the oligopoly of certificate providers that issue digital certificates to clients facing identical independent risks. We found the prices in the equilibrium, and we proof that certificate providers that offer better revocation information are able to impose higher prices to their certificates without sacrificing market share in favor of the other oligarchs. In addition, we show that our model is able to explain the actual tendency of the SSL market where providers with worst QoS are suffering loses.

Keywords: PKI pricing, SSL certificates, CRLs.

1 Introduction

Nowadays, there is a wide range of technology, products and solutions for securing electronic infrastructures. As with physical access security, the levels of security implemented should be commensurate with the level of complexity, the applications in use, the data in play, and the measurement of the overall risk at stake. A consensus has emerged among technical experts and information managers in government and industry that Public Key Infrastructure (PKI) offers the best feasible solution to these issues. PKI [1] has been a popular, yet often reviled technology since its adoption in the early nineties.

Currently deployed PKIs rely mostly on Certificate Revocation Lists (CRLs) for handling certificate revocation [2]. Although CRLs are the most widely used way of distributing certificate status information, much research

** This work is funded by the Spanish Ministry of Science and Education under the projects CONSOLIDER-ARES (CSD2007-00004), FPU grant AP2010-0244, and TEC2011-26452 "SERVET", and by the Government of Catalonia under grant 2009 SGR 1362.

effort has been put on studying other revocation distribution mechanisms in a variety of scenarios [3, 4]. These studies aim to compare the performance of different revocation mechanisms in different scenarios. However, none of these studies have explicitly modeled the interaction among CAs. In this paper, we model this interaction by using a game-theoretic approach.

With the appearance of novel network environments (e.g VANET or MANET), the quantity of CAs in the SSL certificate market is becoming larger and the market concentration diminishes, but it is not simple to eliminate the oligopoly in the short-term. During the 90s, the certification market, the competition among CAs appears mainly as price competition. In this situation, malignant price competition would be detrimental to the interests of the users and lead to the CA's pay crisis. Facing the situation, the main CAs have begun to change the competitive strategies from basic price competition to price and quality of services (QoS) competition. To provide better QoS, CAs have to improve their revocation service, and specifically the freshness of the CRLs. Users will pay more for a service that issues certificate status information faster. Time-to-revocation metric is visible to costumers by checking the CA's repositories where they publicize the revocation information.

The model of this article deals with an oligopoly of CAs which compete in certificate prices and QoS, and do not know the certificate revocation probability in the next interval for sure. The assumption that the revocation probability is *ex-ante* uncertain is quite logical and intuitive. The number of revoked certificates vary with time and in a manner that cannot predictable with certainty. We show that an uncertain revocation probability introduces a systematic risk that does not decrease by selling more certificates. If CAs are risk averse, this effect relaxes price competition. The equilibrium characteristic of the certification market is found by establishing a price competition model with different QoS. We consider that there are diversities in the certification service quality, and we describe factors that affect the service quality such as the CRL lifetime. By combining the characteristics of the certification market and considering the conveniences of modeling, two key parameters are selected to measure the QoS and a duopoly price competition model with service quality differentiation is established.

2 Related Work

Although PKI has been a widely adopted solution for many years now, very few works have dealt with the impact of the revocation mechanism in the prices CAs offer. Most of the literature [4, 5], intend to optimize the revocation mechanism to minimize the overhead or to improve the reliability. However, the most extended revocation mechanism is still CRL. Authors in [6] analyze the revocation mechanisms based on based on empirical data from a local

network. They conclude that the freshness of the revocation data depends on how often the end entities retrieve the revocation information but the bandwidth cost is high if end entities retrieve the revocation lists often.

Ma *et al.* in [7] propose a series of policies that certification authorities should follow when releasing revocation information. According to this study, a CA should take different strategies when providing certificate services for a new type of certificates versus a re-serving type of certificates. Authors give the steps by which a CA can derive optimal CRL releasing strategies and they prove that a CA should release CRLs less frequently in the case that the fixed cost is higher, the variable cost is higher, the liability cost is lower, or the issued age of certificates is shorter. Similarly authors in [8] authors address the CRL release problem through a systematic and rigorous approach which relies on a mix of empirical estimation and analytical modeling. They propose four different models which seek to exploit the variation in certificate specific properties to provide guidance to the CA in determining the optimal CRL release intervals and the associated costs. However, none of these works neither analyze the impact of CRLs releasing policies in the prices that the CA charges nor model the interaction among CAs. In this paper, we address these issues using a game theoretic approach.

3 Modeling the Certificate Provider Competition

To formalize our arguments we describe a model of the certificate market with profit-maximizing certification authorities and a continuum of network users. When a user requests the status of given certificate, the CA does not always provides the most updated information but a pre-signed CRL [4, 5]. In this context, the CA will bear the liability cost due to any damage that may occur between the revocation of a certificate and the release of the CRL.

3.1 Demand for certificates

We consider an oligopoly of A CAs, indexed by $i = 1, \cdots, A-1$, and N users in the economy, where N is large relative to A.Each user has the same strictly concave expected utility function and faces the risk to lose l when using a revoked certificates. The probability π of operating with a revoked certificate is equal for each user in the network, and conditional on π operating with revoked certificates of different users are statistically independent. This probability is out of the user's control so that no moral hazard problem arises. Except for their probabilities of operating with revoked certificates, individuals are assumed to be identical. However, π is not known *ex-ante* with certainty but is a random variable distributed on $[\underline{\pi}; \overline{\pi}]$ with cumulative

density function $F(\pi)$. Each user has an initial wealth $w > 0$. When operating with a revoked certificate, users may suffer a loss. We assume that the user's wealth exceeds the potential loss, that is, $l \leq w$.

Users can purchase different certificate types from the CA with different revocation updating service. We characterize this product by the price of the certificate $P_i > 0$ and an indemnity $C_i > 0$ the CA pays to the user if it suffers from an attack and operates with another user whose certificate was revoked. Note that as CRLs are not issued each time a certificate is revoked but periodically, users will be operating with outdated information. Let (P_i, C_i, t_i, s_i) be a certificate contract offered by CA_i which specifies the price P_i to be paid by an user and the level of coverage C_i paid to the user if an attack takes place and she operates with a revoked certificate. Let t_i represent the CRL updating interval, and s_i represent the security level.

Let us assume that the total utility U which users can get after they purchase a certificate consists of two parts. The first part is wealth utility which represented by U_w the other part is QoS utility which the applicant can get after they obtained the CA's services, represented by U_{QoS}. The total utility U is defined as:

$$U(P_i, C_i, t_i, s_i) = \alpha_1 U_w + \alpha_2 U_{QoS}, \forall \alpha_k \in [0,1] \text{ and } \sum \alpha_k = 1;\ k = 1, 2. \tag{1}$$

where α_i represents the significance level of U respectively.

On the one hand, we calculate the wealth utility. If no attack due to misuse of a revoked certificate happens after the user has purchase the service the CA, a user gains $w - P_i$, on the contrary a user gains $w - P_i + C_i$. We assume that all users have same loss with two-point distribution:

$$\mu = (w - P_i)(1 - \pi) + (w - P_i + C_i)\pi \ = w - P_i + \pi C_i, \tag{2}$$

$$\sigma^2 = \pi(1 - \pi)C_i^2. \tag{3}$$

Hence we can characterize the wealth utility by the mean and variance of Eq. (2) and Eq. (3) respectively. Thus, we can define U_w as a mean-variance utility function:

$$U_w(P_i, C_i) = \mu - R\sigma^2, \tag{4}$$

where R represents the Arrow-Pratt index of absolute risk aversion. This means that the larger R is, the more risk averse the user is and the smaller U_w is.

On the other hand, let U_{QoS} be a linear function of the QoS that the CA offers. Thus, we define U_{QoS} as:

$$U_{QoS}(t_i, s_i) = \pi\theta \left(\beta_1 s_i + \beta_2 \frac{1}{t_i}\right), \forall \beta_k \in [0,1], \sum \beta_k = 1 \text{ and } \theta > 0;\ k = 1, 2. \tag{5}$$

where θ represents the quality preference parameter of the user, and β_1 represents the user's preference to security level and β_2 represents the user's

preference to CRL issuing interval. Note that the higher the level of security the CA provides, the larger U_{QoS} is; the longer the CRL updating interval is, the smaller U_{QoS} is. It is also worth noting that θ is unknown to the CAs a priori.

In order to calculate the total utility of the user, we must unify the dimension of the security level and the CRL updating interval. Thus, using (1),(4) and (5) the total utility is calculated as:

$$U(P_i, C_i, t_i, s_i) = \alpha_1[w - P_i - \pi C_i - R\pi(1-\pi)C_i^2] + \alpha_2\left[\pi\theta\left(\beta_1 s_i + \beta_2 \frac{1}{t_i}\right)\right]. \quad (6)$$

Note that according to this expression, users are willing to pay higher prices for those certificates whose issuer provides a better QoS. Note that issuing certificate status information faster, highly increases the QoS of the revocation service. Thus, certificates linked to a better revocation service provide more utility to the user.

3.2 Supply of certificates

We consider an oligopoly of CAs operating in the certification market. CAs compete for users by offering certificates and CRLs. The service qualities of their CA products are also different. The level of service quality is mainly shown by the CRL updating interval and the security level[1].

When choosing a CA, a user takes into account several factors. Our goal is to gauge the impact of the revocation service on the certificate prices. However, it should be noted that, for convenience, many website owners choose the registrar's authority regardless of the price. Before issuing a certificate, the CA verifies that the person making the request is authorized to use the domain. The CA sends an email message to the domain administrator (the administrative or registrant contact, as listed in the Whois database) to validate domain control. If there is no contact information in the Whois database or the information is no longer valid, the customer may instead request a Domain Authorization Letter from his/her registrar and submit the letter to the CA as proof of his/her domain control. If the administrative/registrant contact fails to approve the certificate request, the request is denied. This authentication process ensures that only an individual who has control of the domain in the request can obtain a certificate for that domain. Therefore as CAs compete by quoting a certificate price which has associated a particular quality of service, we have Bertrand competition. The CA that quotes the lowest certificate price with the highest QoS sells to all users.

[1] Note that additional QoS parameters could be introduced in the model. In fact, CAs distinguish themselves by offering additional value-added services (e.g. GoDaddy bundling domain registration with certificate issuance), turn-around time, etc.

4 Equilibrium Certificate Providers

In this section we consider the certification industry with an oligopoly of A certification authorities and analyze the competitive forces that determine equilibrium of certificate selling. Our main goal is to find the prices at which CAs obtain their maximum profit, i.e., when they reach the game equilibrium. Recall that these certificates differ in the QoS so that $\forall i, j; i \neq j, t_i \neq t_j$ and $s_i \neq s_j$. We assume that the certification market is covered in full. Users will intend to maximize their utility, i.e.:

$$\theta^* = \arg\max_{\theta} \quad U(P_i, C_i). \tag{7}$$

On the other hand, CAs will intend to minimize their costs. The CA's costs consists of fixed and variable costs. Each time a new CRL is issued, a CA incurs both fixed and variable costs. The fixed cost depends on two factors. The fix component is due to the release of a new CRL, and does not depend on the number or certificate type. The variable factor depends on the number of certificates contained in the CRL (i.e. depends on the size of the CRL) and on the type of certificate (i.e. certificate with higher security level induce higher costs). Note that in this variable cost it is included the cost of processing each certificate revocation request. We define the service quality cost of CA_i (i.e. $Q(s_i, t_i)$) as a variable that includes both fixed and variable costs associated to the QoS. The first and second derivative of $Q(s_i, t_i)$ with respect to s_i, t_i are positive. Hence, we can calculate the gain function G_i of any CA_i:

$$G_i = \theta^* P_i - Q(s_i, t_i), \tag{8}$$

where the gain function captures the overall profits of CA_i for a given certificate product characterized by (P_i, C_i).

We assume that the game between the two CAs is static with incomplete information, they choose the respective certificate price at the same time to maximize their profits. Now we differentiate (8) with respect to P_i and C_i. In order to obtain the certificate price and the coverage in the equilibrium, let each derivative formula equal to zero. Solving the resulting linear system, we will obtain the price of each CA P_i^* and the corresponding coverage C_i^*.

$$P_i^* : \frac{\partial G_i}{\partial P_i} = 0, \quad C_i^* : \frac{\partial G_i}{\partial C_i} = 0. \tag{9}$$

4.1 Duopoly of CAs

To better illustrate the results obtained in the previous section, we particularize the case of the oligopoly to a duopoly where only two CAs are offering

certificates. This simplification, we allows us to draw some conclusion that can be easily extrapolated to the real scenario where there are more than a dozen CAs. To show that the level of service quality depends on the CA, we assume that the CA indexed by $i = 1$ offers better quality than the second CA in both QoS parameters, i.e., $t_1 < t_2$ and $s_1 > s_2$.

Following the methodology aforementioned, we have to find the prices in the equilibrium. In this situation, first we find the value of θ^* at which a user has no obvious trend between the certificates offered by different CAs.

$$\alpha_1[w - P_1 - \pi C_1 - R\pi(1-\pi)C_1^2] + \alpha_2\left[\pi\theta\left(\beta_1 s_1 + \beta_2\frac{1}{t_1}\right)\right] = \alpha_1[w - P_2 - \pi C_2 - R\pi(1-\pi)C_2^2] + \alpha_2\left[\pi\theta\left(\beta_1 s_2 + \beta_2\frac{1}{t_2}\right)\right], \tag{10}$$

which results in:

$$\theta^* = \frac{\alpha_1\left(P_1 - P_2 + \pi C_1(1 + RC_1 - R\pi C_1) - \pi C_2(1 - RC_2 + R\pi C_2)\right)}{\pi\alpha_2 K} \tag{11}$$

where $K = \beta_1(s_1 - s_2) + \beta_2\left(\frac{1}{t_1} - \frac{1}{t_2}\right)$. So the market demand of CA_2 is θ^*, and the demand of CA_1 is $1 - \theta^*$.

Using (8) we calculate the gain function G_i of CA_1 and CA_2 :

$$G_1 = (1 - \theta^* P_1) - Q(s_1, t_1), \tag{12}$$

$$G_2 = \theta^* P_2 - Q(s_2, t_2). \tag{13}$$

We obtain the certificate price and the coverage in the equilibrium :

$$P_1^* = \frac{2\pi\,\alpha_2 K}{3\alpha_1} \qquad P_2^* = \frac{\pi\,\alpha_2 K}{3\alpha_1}, \qquad C_1^* = C_2^* = \frac{1}{2R\,(-1+\pi)}. \tag{14}$$

From these results we can conclude that:

- In the equilibrium, when both CAs achieve their maximum gain, CA_1 obtains a higher price than CA_2. This is mainly due to the fact that when both CAs have associated the same probability of an attack, as the QoS of the first CA is better so that CA_1 can set a higher price per certificate.
- In the equilibrium, the coverage that each CA should establish is the same and is inversely proportional to the risk-aversion and the probability of operating with a revoked certificate.

5 Analysis and Results

5.1 Impact of the preference ratio $\frac{\alpha_2}{\alpha_1}$

As the ratio between the preference of QoS utility and wealth utility of the user increases (i.e., users are more interested in a high security service and

a good revocation mechanism) the prices of both CAs in the equilibrium also increase. This effect is reasonable, as the improvement of the revocation mechanism gives a higher security level which also increases the costs. This cost increment is compensated with a higher price in the equilibrium. Analyzing two CAs operating in the oligopoly such that $t_i < t_j$ and $s_i > s_j$, it is worth noting that the increment speed of CA_i's QoS is faster than that of CA_j, so the increment speed of its certificate price is also faster than the other CA.

5.2 Impact of the security level difference

When the level of security that a CA offers is much higher than in the others, the certificate value is also much higher. Thus, CAs that offer certificates with higher level of encryption and larger keys are able to make their certification product differentiable. For instance, SSL security levels vary depending upon the way on SSL certificate is installed on a server and the configuration used. SSL is simple to use but its security can be compromised if basic installation an configurations are not completed to a competent level, hackers are then able to decrypt the security on a badly installed SSL certificate. Once the certificates of a CA are differentiable from the other CAs, CAs do not have to use malignant prices anymore to compete. As the difference of this QoS between CAs becomes bigger, the prices that they can charge also increase. Note that if the preference extent which the user shows to the security level (i.e. β_1) increases, the differences in the certificates as products will be more apparent, thus the increase in the CA's certificate prices will also increase. The same results are expected with the increment of the interest of the users to a better service from the CAs (α_2), that is, not higher security but also a more efficient revocation mechanism.

5.3 Impact of the QoS of the revocation mechanism

CAs that are able to offer revocation mechanisms with fresher information and high availability are able to make their certification product differentiable. Recall that this QoS increase of the revocation mechanism induces higher costs, as revocation information has to be issued more frequently. These costs are compensated with an increase of the price that CAs can charge for the certificates in the equilibrium. The reasons are the same that in the previous case, but now users pay more attention to the revocation mechanism rather than to the level of security. Analytically, that means that β_2 increases, so that the user is more interested in the efficiency of the revocation mechanism. This increase induces a proportional increase in the equilibrium prices of the CAs. Note that in this case, the increase of CA_i

which has higher QoS of the revocation mechanism is faster than that of CA_j. Again, the CA that has better service (no matter if it is higher security level or a more efficient revocation mechanism) has the advantage in competition.

5.4 Impact of the revocation probability

Logically, with an increase of the probability of operating with a revoked certificate, CAs charge more for their certificates. The reason is obvious as the CAs set they price mainly based on a forecast of this probability. An increase of π will induce an increase of the compensation expenses that a CA will have to pay to any victim of an attack due to the misuse of a revoked certificate. Consequently, this increase will lead to a proportional increase of compensation cost and service cost so that the CAs have to increase their prices to compensate the cost increases. Note that this increase is twice faster in the case of the CA_i.

6 Case Study: SSL Providers

Finally, to corroborate the benefits of the presented model, we analyze the case of current SSL providers that issue digital certificates. An SSL certificate can be obtained from amounts as low as $43 to as high as $3000 per year. Whilst the type of encryption can be the same, the cost is determined by the rigour of the certification process as well as the assurance and warranty that the vendor can provide. Table 1 shows the prices and QoS that the leading CAs operating in the SSL Certificate market are offering. The SSL Certificate market was traditionally dominated by a small number of players, namely VeriSign and Thawte. Whilst in a monopolistic position they had the capability of charging inflated prices for a commodity product. However new providers with no necessity to hold prices high were able to offer SSL certificates at far more reasonable prices.

The SSL certificate vendors provide insurance against the misuse of certificates and this differs from one vendor to another. Verisign provides warranties of up to $250,000 while Entrust and GoDaddy offer a $10,000 warranty. The higher the insurance, the more inscription/authentication is provided by the SSL vendors. Analyzing Table 1, it is worth noting that not always a lower price means lower quality. Therefore, it is evident that current CAs operating in this market are competing both in price and quality of service.

To test whether these factors are determinant factors for the certficate prices, we perform a multivariate regression analysis explaining the yearly price of SSL certificates. General regression investigates and models the relationship between a response (Certificate price) and predictors (Warranty, issuing interval and CRL lifetime). Note that the response of this model is

SSL Provider	Product Name	Price/Year($)	Warranty($)	Assurance	Mean Issuing time	Mean CRL lifetime
COMODO	EnterpriseSSL Platinum	311.80	1,000,000	High	Under 1 hour	4 days
COMODO	InstantSSL Pro	169.80	100,000	High	Under 1 hour	4 days
Verisign	Secure Site Pro Cert	826.67	2,500,000	High	2-3 days	15 days
Verisign	Managed PKI for SSL Std	234.00	100,000	High	2-3 days	15 days
GeoTrust	QuickSSL Premium	118.00	100,000	Low	Immediate	10 days
GeoTrust	True BusinessID	159.20	100,000	High	2 days	10 days
Go Daddy	Standard SSL	42.99	10,000	Low	Immediate	1 day
Go Daddy	Standard Wildcard	179.99	10,000	Low	Immediate	1 day
Entrust	Advantage SSL Certificates	167.00	10,000	High	2 days	1 week
Entrust	Standard SSL Certificates	132.00	10,000	High	2 days	1 week
Thawte	SSL 123	129.80	-	Low	Immediate	1 month
Thawte	SGC Super cert	599.80	-	High	2 days	1 month

Table 1. SSL Certificate Types and Services offered by main CAs [9].

continuous, but you we have both continuous and categorical predictors. You can model both linear and polynomial relationships using general regression. With this model we determine how the certficate price changes as a particular predictor variable changes. We use data from a survey of CAs performed in 2010 [9]. The obtained regression model is expressed in the following equations for high and low assurance certificates, respectively:

$$Price/Year(\$) = 98,4353 + 0,000220857\, W - 0,549141\, \overline{I_{time}} + 8,6116\, \frac{1}{\overline{CRL_{Lf}}},$$

$$Price/Year(\$) = 20,0405 + 0,000220857\, W - 0,5491411\, \overline{I_{time}} + 8,6116\, \frac{1}{\overline{CRL_{Lf}}},$$

where W denotes the warranty, $\overline{I_{time}}$ is the mean issuing time, and $\overline{CRL_{Lf}}$ is the mean lifetime of the CRLs issued by the CA.

Note that both regression equations show that the coefficient of the predictor associated to the CRL's mean lifetime is significant. In fact, the p-value associated to this predictor is $0,008$ which indicates that is statistically significantly. Overall, the variables within the model are explaining a large portion of the variation in the certificate price. With a coefficient of determination R^2 above the 81%, we are capturing important drivers of certificate prices. The residuals from the analysis are normally distributed, i.e., no evidence of nonnormality, skewness, or unidentified variables exists.

Using the proposed model, we are able to explain these different prices and the corresponding market share and they potential evolution. First we analyze the number of revoked certificates as it will determine the probability of operating with a revoked certificate. Figure 1 shows the evolution of the daily number of revoked certificates per CA. These data were collected from different SSL CRLs that the CAs make public at their repositories. It is worth

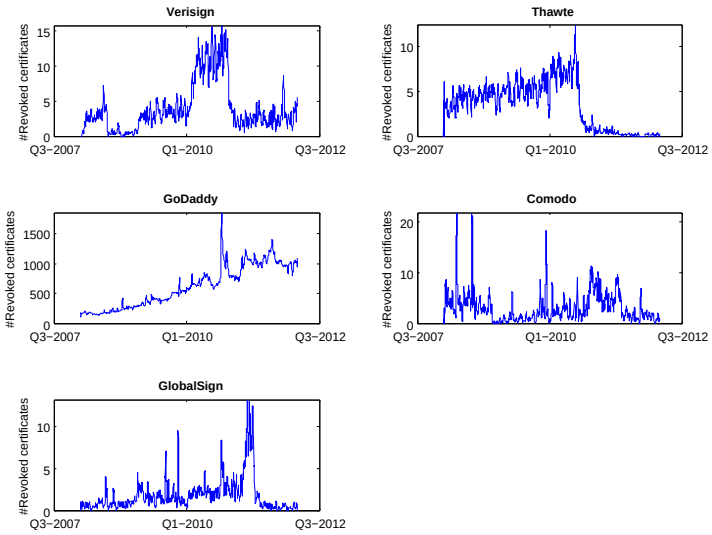


Fig. 1. Evolution of the daily number of revoked certificates per CA.

noting, that the number of revoked certificates highly varies depending on the CA. Thus, GoDaddy revokes more than 500 certificates per day on average while VerSign revokes less than 4 certificates per day on average. Therefore, the probability π of operating with revoked certificates is higher when trusting certificates issued by GoDaddy. As our model shows, using expression (14), the probability π directly affects the price of the certificate. Thus, as GoDaddy has a higher π, we would expect to charge less for its certificates. However, the price is quite similar to its competitors. Thus, GoDaddy is not able to sell as much certificates as the other oligarchs, and its market share is smaller.

Our model would expect GoDaddy to compete not only in prices but also in QoS to gain market share. As our model shows, the reaction of GoDaddy to compete in the oligopoly is to offer better quality of service. From table 1, we can see that GoDaddy is the CA that issues CRLs more often. Using this CRL releasing policy, users increase their utility and, at the same time, the probability of operating with a revoked certificate is also reduced. However, the variable costs increase due to this way of issuing CRLs. Similarly, Comodo intends to gain market share by decreasing the time it takes to issue a certificate and also reducing the CRL lifetime. Note that VeriSign, the leading CA, is the one who is offering the worst QoS, both in terms of CRL lifetime and time to issue a new certificate.

7 Conclusions

The market of certificate providers can be described as an oligopoly where oligarchs compete not only in price but also in quality of service. In this paper we have modeled this oligopoly using a game theoretic approach to find the prices in the equilibrium. We have been able to capture the QoS of the products offered by a CA, by means of the timeliness of the revocation mechanism and the security level. In our model of the certification industry with profit-maximizing CAs and a continuum of individuals we showed that although the undercutting process in certification prices seems similar to the price setting behavior of firms in Bertrand competition there exists a crucial difference depending onf the QoS of the revocation service. The solution of the game for two CAs in the oligopoly that offer certificates with different QoS shows that the revenues of the CA which provides a better revocation mechanism and a higher security level are larger. Therefore, a CA when setting the prices of its certificate and the compensation expenses, it has to take into account not only the probability of operating with a revoked certificate, but also the quality of the revocation mechanism and the security level. Thus, any CA should comprehensively consider the difference in quality of its services compared with other CAs.

References

1. C. Adams and S. Farrell. Internet X.509 Public Key Infrastructure Certificate Management Protocols. RFC 2510, Internet Engineering Task Force, March 1999.
2. R. Housley, W. Polk, W. Ford, and D. Solo. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. RFC 3280, Internet Engineering Task Force, April 2002.
3. T. Perlines Hormann, K. Wrona, and S. Holtmanns. Evaluation of certificate validation mechanisms. *Comput. Commun.*, 29:291–305, February 2006.
4. A. Arnes. Public key certificate revocation schemes. 2000. Queen's University. Ontario, Canada. Master Thesis.
5. D.A. Cooper. A more efficient use of Delta-CRLs. In *2000 IEEE Symposium on Security and Privacy. Computer Security Division of NIST*, pages 190–202, 2000.
6. Mona H. Ofigsbø, Stig Frode Mjølsnes, Poul Heegaard, and Leif Nilsen. Reducing the cost of certificate revocation: a case study. In *Proceedings of the 6th European conference on Public key infrastructures, services and applications*, EuroPKI'09, pages 51–66, Berlin, Heidelberg, 2010. Springer-Verlag.
7. Chengyu Ma, Nan Hu, and Yingjiu Li. On the release of CRLs in public key infrastructure. In *Proceedings of the 15th conference on USENIX Security Symposium - Volume 15*, Berkeley, CA, USA, 2006.
8. Nan Hu, Giri K. Tayi, Chengyu Ma, and Yingjiu Li. Certificate revocation release policies. *J. Comput. Secur.*, 17:127–157, April 2009.
9. WhichSSL. SSL Market Share, 2010. [Online] http://www.whichssl.com/ssl-market-share.html.

On the self-similarity nature of the revocation data

Carlos Gañán, Jorge Mata-Díaz, Jose L. Muñoz
Oscar Esparza and Juanjo Alins

Universitat Politècnica de Catalunya, Telematics Department, Barcelona (Spain)
{carlos.ganan,jmata,jose.munoz,oscar.esparza,juanjo}@entel.upc.edu

Abstract. One of the hardest tasks of a Public Key Infrastructure (PKI) is to manage revocation. Different revocation mechanisms have been proposed to invalidate the credentials of compromised or misbehaving users. All these mechanisms aim to optimize the transmission of revocation data to avoid unnecessary network overhead. To that end, they establish release policies based on the assumption that the revocation data follows uniform or Poisson distribution. Temporal distribution of the revocation data has a significant influence on the performance and scalability of the revocation service. In this paper, we demonstrate that the temporal distribution of the daily number of revoked certificates is statistically self-similar, and that the currently assumed Poisson distribution does not capture the statistical properties of the distribution. None of the commonly used revocation models takes into account this fractal behavior, though such behavior has serious implications for the design, control, and analysis of revocation protocols such as CRL or delta-CRL.

Keywords: Self-similarity, Certification, Public Key Infrastructure, Revocation.

1 Introduction

Today we are in the midst of an electronic business revolution. It is of utmost importance that mechanisms are set up to ensure information and data security. Organizations have recognized the need to balance the concern for protecting information and data with the desire to leverage the electronic medium. Public Key Infrastructure (PKI) is a step toward providing a secure environment by using a system of digital certificates and certificate authorities (CAs). However, one of the most important aspects in the design of a PKI is certificate revocation.

Certificate revocation is the process of removing the validity of a certificate prematurely. There could be multiple reasons for revoking a certificate; such as the certificate holder leaves the organization or there is

a suspicion of private key compromise. When a certificate is revoked, the information about the revoked certificate needs to be published. Some of the methods that a CA can use to revoke certificates are:

- Periodic Publication Mechanisms: Information about revoked certificates can be posted on a certificate server so that the users are warned from using those certificates. This mechanism includes the use of Certificate Revocation Lists (CRL) and Certificate Revocation Trees (CRT). A CRL is a signed list of certificates that have been revoked or suspended. CRT is a revocation technology, which is based on Merkle hash trees, where the tree represents all known certificate revocation information relevant to some known set of PKI communities.
- Online Query Mechanisms: Online Query Mechanisms comprise Online Certificate Status Protocol (OCSP) and Online Transaction Validation Protocols. OCSP is used to obtain online revocation information about certificates, and Online Transaction Validation Protocols are used for online validation, such as business transactions through credit cards.

A revocation method is selected by an organization based on the cost, infrastructure, and volumes of transactions that are expected. To gauge these costs, different revocation mechanisms are tested under the assumption that the revocation events follow a specific probability distribution. Most theoretical frameworks and simulation studies for performance evaluation assume that the temporal distribution of queries follows a Poisson distribution. Thus, organizations estimate the infrastructure needed to deploy the PKI and the associated costs. However, in this article, we demonstrate that revocation data is statistically *self-similar*, that none of the commonly used revocation models is able to capture this fractal behavior, and that such behavior has serious implications for the design, control, and analysis of revocation protocols such as CRLs.

We start by analyzing the validity of Poisson-like process assumption. We use publicly available CRLs from different certification authorities (containing more than 300,000 revoked certificates over a period of three years). Our analysis demonstrates that the Poisson distribution fails to capture the statistical properties of the actual revocation process. We also see that the Poisson distribution grossly under-estimates the bandwidth utilization of the revocation mechanism. At first glance, this might look like an obvious result, since after all as a memoryless process, Poisson distribution cannot be expected to model periodic trends like daily, weekly and monthly cycles in revocation rates. We show however that the modeling inability transcends simple cycles. In particular, we will show that

self-similarity has a severe detrimental impact on the revocation service performance.

Results of our analysis, including burstiness at all scales, strongly suggest self-similar nature of revocation events. We confirm this by estimating the Hurst parameter for the observed distribution and showing that the estimates validate self-similar nature of the revocation lists. Beyond invalidating Poisson-like distributions, this proof of self-similarity has the important implications on CA utilization, throughput, and certificate stratus checking time. Intuitively, as the revocation process is bursty (non-uniformly distributed) the CA will be partially idle during low burst periods and vice versa. Thus, the revocation lists will grow non-uniformly, and current updating policies will result bandwidth inefficient.

The rest of this article is organized as follows. Section 2 gives the necessary statistical background required to understand self-similar processes and long range dependency. In Section 3, we discuss the methodology we used to collect and analyze real-world revocation data. We demonstrate self-similar nature of the revocation data, followed by a Hurst parameter estimation. In Section 4 we discuss how the observed self-similarity has crucial implications on performance of the revocation service. Next section discusses the related work in the area. Finally, we conclude in Section 6.

2 Background

2.1 Self-Similar Processes

A phenomenon which is self-similar looks the same or behaves the same when viewed at different degree of magnification. Self-similarity [1] is the property of a series of data points to retain a pattern or appearance regardless of the level of granularity used and can be the result of long-range dependence (LRD) in the data series. One of the main properties of the self-similar data is burstiness [1]. Bursty data do not possess a stable mean value. Significant differences in the mean value are one of the reasons why bursty data are more difficult to control than shaped one. If a self-similar process is bursty at a wide range of timescales, it may often exhibit long-range dependence. Long-range-dependence means that all the values at any time are correlated in a positive and non-negligible way with values at all future instants.

A stochastic process $Y(t)$ is *self-similar* with Hurst parameter H if for any positive stretching factor d, the distribution of the rescaled and

reindexed process $d^{-H}Y(dt)$ is equivalent to that of the original process $Y(t)$. This means for any sequence of time points $t_1, \ldots, t_n$ and any positive constant d, the collections $\{d^{-H}Y(dt_1), \ldots, d^{-H}Y(dt_n)\}$ and $\{Y(t_1), \ldots, Y(t_n)\}$ are governed by the same probability law. When the values of H are in the interval $(0.5, 1)$, the process presents LRD. A value of H equal to 0.5 indicates the absence of LRD. This means that the smoothing with aggregation is much slower for self-similar processes, the greater the degree of self-similarity, the slower will be smoothing with aggregation.

Three implications of self-similarity are:

- No natural length of bursts.
- Presence of bursts in all time scales.
- Process does not smooth out on aggregation.

2.2 Statistical Tests For Self-Similarity

The practical way to estimate degree of self-similarity is to measure the values of Hurst exponent. In this paper we use five methods to test for self-similarity (details about these methods are described in [2, 3]).

The first method, the variance-time plot, relies on the slowly decaying variance of a self-similar series. The variance of $Y^{(m)}$ is plotted against m on a log-log plot; a straight line with slope (β) greater than -1 is indicative of self-similarity, and the parameter H is given by $H = 1 - \beta/2$. The second method, the R/S plot, uses the fact that for a self-similar dataset, the rescaled range or R/S statistic grows according to a power law with exponent H as a function of the number of points included (n). Thus the plot of R/S against n on a log-log plot has slope which is an estimate of H. The third approach, the periodogram method, uses the slope of the power spectrum of the series as frequency approaches zero. On a log-log plot, the periodogram slope is a straight line with slope close to the origin.

While the preceding three graphical methods are useful for exposing faulty assumptions (such as non-stationarity in the dataset) they do not provide confidence intervals. The fourth method, called the Whittle estimator does provide a confidence interval, but has the drawback that the form of the underlying stochastic process must be supplied. The two forms that are most commonly used are fractional Gaussian noise (FGN) with parameter $1/2 < H < 1$, and Fractional ARIMA(p,d,q) with $0 < d < 1/2$ (for details see [2])). These two models differ in their assumptions about the short-range dependences in the datasets; FGN assumes no short-range

dependence while Fractional ARIMA can assume a fixed degree of short-range dependence. There are several other methods in frequency and time domain to measure the Hurst parameter.

Finally, we use the Detrended Fluctuation Analysis (DFA) [4], which aims to highlight the long-range dependence of a time series with trend. DFA method is a version for time series with trend of the method of aggregated variance used for a long-memory stationary process. It consists in aggregating the process by windows with fixed length, detrending the process from a linear regression in each window, computing the standard deviation of the residual errors (the DFA function) for all data, and finally, estimating the coefficient of the power law from a log-log regression of the DFA function on the length of the chosen window.

3 Examining the self-similarity of the revocation process

3.1 Data Collection

In order to capture the temporal correlation of the revocation process, first we have to gather a large sample of revocation data. The approach we follow consists in collecting revocation data from different certification authorities using their available CRLs. In particular, we built some scripts to download and preprocess the CRLs from the following CAs[1]: VeriSign, GoDaddy, Thawte, and Comodo.

Issuer Name	Number of Revoked Certificates	Last Update	Next Update
GoDaddy	932,900	2012/02/01	2012/02/03
VeriSign	5,346	2012/02/02	2012/02/16
Comodo	2,727	2012/02/03	2012/02/06
GlobalSign	7,591	2012/02/02	2012/03/03
Thawte	8,061	2012/02/01	2012/02/16

Table 1: Description of the collected CRLs.

Though we concentrate our analysis on CRL because it is the most common and simplest method for certificate revocation [6], we expect the

[1] According to NetCraft's survey [5], using these CAs we cover most of the world market for SSL.

captured pattern to be extensible to any other revocation mechanism (e.g. OCSP).

Once downloaded the revocation data, we preprocess these data to remove duplicated information (e.g. certificates that are revoked due to several reasons). Note that when a revoked certificate expires, it typically remains in the CRLs for one additional publication interval, so we preprocess the CRLs to remove expired certificates too. In this sense, Thawte's and GlobalSign's CRLs may contain duplicate entries for the same certificate because of their policy statements. These policy statements impose that a certificate that is revoked by several reasons must be included in the CRL as many times as the number of revocation reasons. Thus, we remove any duplicate entry from the composite dataset, and tally the number of revocations per day. Finally, we build a dataset that covers non-expired revoked certificates from 2008 to 2012 (see Figure 1).

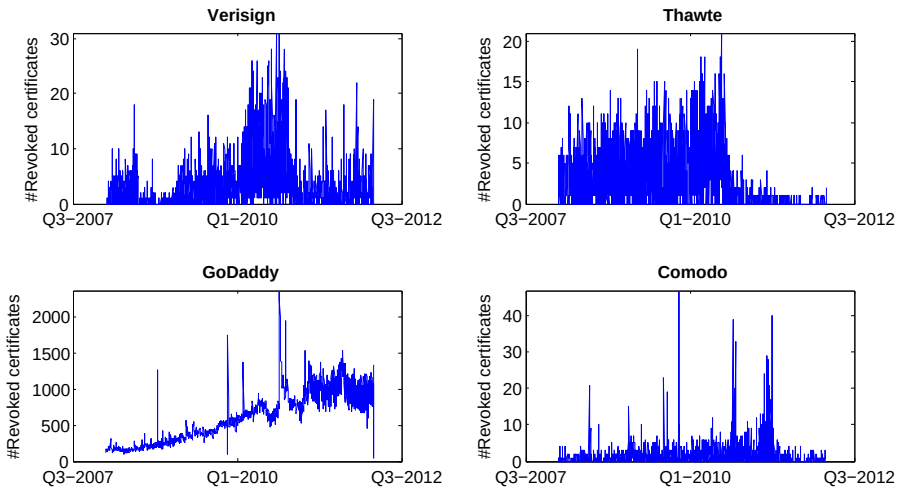


Fig. 1: Number of daily revoked certificates evolution for each CA.

3.2 Evidence of Burstiness

Before providing formal estimation of self-similarity, we provide a graphical evidence of bursty nature of the revocation data at different time scales. We also show that this observed burstiness is not accounted by the Poisson distribution. In Figure 2, we show the revocation logs in four different time scales-ranging from 1 hour to 1 day. Each plot is obtained by

changing the time resolution. In Figure 2 we can observe different evident trends; (i) Burstiness in all time scales: the burstiness of the revocation process does not disappear when changing the time scales. (ii) Lack of natural length of bursts: The figure shows burstiness ranging from days to months. Note that the full duration of the figure with the largest time slot is 1,000 days, and some of the bursts have many hours of duration.

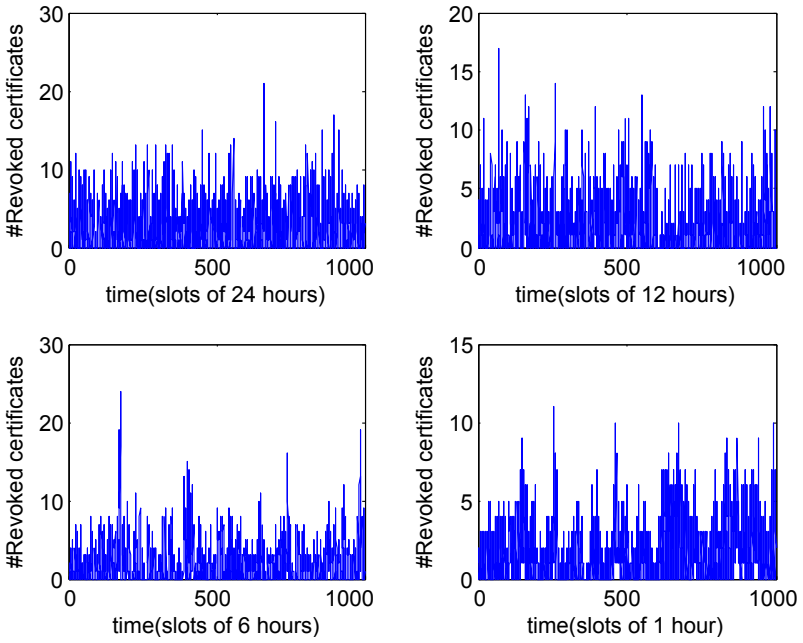


Fig. 2: Revocation Bursts over Four Orders of Magnitude.

In addition, it is worth noting the difference between this bursty pattern and a Poisson process. A Poisson process smooths out with large time scales and resembles a uniformly distributed white noise at higher time scales. In contrast to the revocation process, in a Poisson process the burstiness vanishes in coarse time scales, longer length bursts are absent, and bursts smooths out much faster. Thus, the trends of self-similarity present in the revocation data discussed above are totally absent for Poisson processes.

Therefore, modeling the revocation process as Poisson is clearly inadequate, and is thus likely to give unrealistic results. We will elaborate this

analysis in the next section, and discuss the consequences of self-similarity in the following sections

3.3 Statistical Analysis of Self-Similarity

In this section, we use five different methods to estimate the Hurst parameter to demonstrate the long range dependency of the revocation events formally. Since there are different manifestations of self-similarity, different methods in time and frequency domains are used in practice for the estimation (see Sec. 2.2). Note that when using these estimators with real-life revocation data containing noise, cycles and trends, they might estimate different values of the Hurst parameter. For that reason, we use multiple methods, report the correlation coefficients and confidence intervals by different methods, and visually inspect the data for trends and cycles. The chances of estimates agreeing on real data is small [7], but if most of the estimates are above 0.5 the LRD is likely to exist.

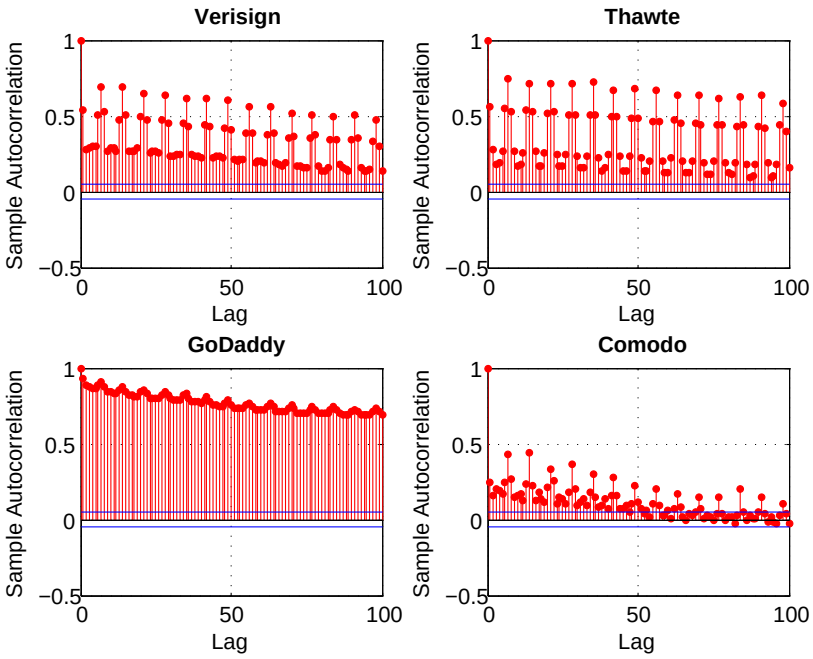


Fig. 3: Autocorrelation function of the revocation process per CA.

First of all, we start analyzing the autocorrelation of the revocation data. Recall that in a self-similar process autocorrelations decay hyperbolically rather than exponentially fast, implying a nonsummable autocorrelation function $\sum_k r(k) = \infty$ (long-range dependence). For the frame data, the empirical autocorrelation functions $r(k)$ are shown in Fig. 3, with lag k ranging from 0 to 100. Notice that $r(k)$ decreases slower than exponentially no matter the CA. The curve does decay toward zero, but it does so extremely slowly. The very slowly decaying autocorrelations are indicative of LRD.

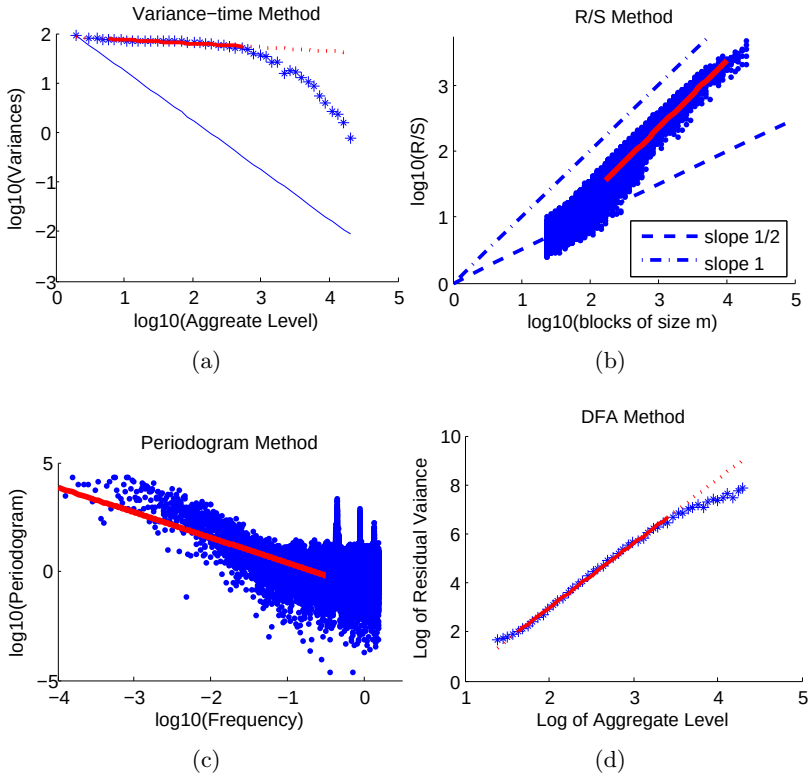


Fig. 4: Graphical methods for checking for self-similarity of the revocation process from GoDaddy (a) variance-time plot, (b) pox plot of R/S, (c) periodogram plot, and (d) DFA plot.

In the following, we use five different methods for assessing self-similarity described in Section 2.2: the variance-time plot, the rescaled range (or

R/S) plot, the periodogram plot, the DFA plot and the Whittle estimator. We concentrated on individual months from our revocation time series, so as to provide as nearly a stationary dataset as possible. To provide an example of these approaches, analysis of a single month from GoDaddy revocation data is shown in Figure 4. The figure shows plots for the four graphical methods: variance-time (upper left), rescaled range (upper right), periodogram (lower left) and DFA (lower right). The variance-time plot is linear and shows a slope that is distinctly different from -1 (which is shown for comparison); the slope is estimated using regression as -0.077, yielding an estimate for H of 0.96. The R/S plot shows an asymptotic slope that is different from 0.5 and from 1.0 (shown for comparison); it is estimated using regression as 0.95, which is also the corresponding estimate of H. The periodogram plot shows a slope of -0.14 (the regression line is shown), yielding an estimate of H as 0.83. Finally, the Whittle estimator for this revocation data (not a graphical method) yields an estimated Hurst value of 0,923 with a 95% confidence interval of $(0.87, 0.95)$.

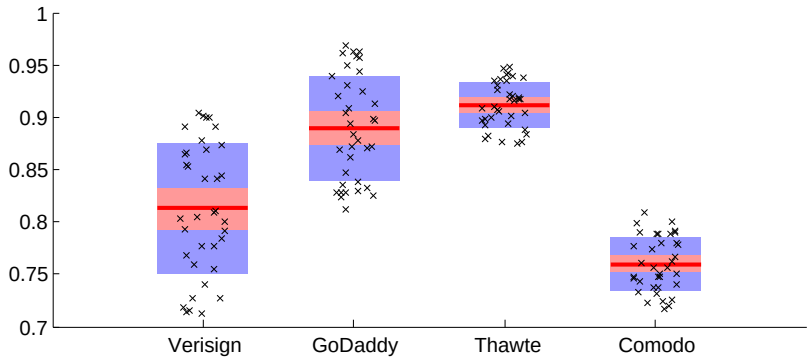


Fig. 5: Summary plot of estimates of the Hurst parameter H for all the CAs.

Once we have seen that GoDaddy presents a significant self-similar pattern, we analyze the rest of the CAs. To that end, we use the whittle estimator to obtain the Hurst value per CA and month. We chose this estimator because it gives more refined measurement than other estimation techniques and it provides confidence levels for the Hurst parameter [8]. Note that we are not interested in estimating the exact value of the Hurst parameter but to prove the existence of self-similarity in the revocation

data. Figure 5 shows the H parameter of each CA and the 95% confidence interval. It is worth noting that depending on the month there are some CAs whose H parameter varies significantly. However, no matter neither the CA nor the month, the Hurst value is always above 0.7. This means that the revocation process of any CA presents LRD.

4 Significance of self-similarity for revocation data management

Our collected data from real CAs show dramatically different statistical properties than those assumed by the stochastic models currently considered in the literature. Almost all these models are characterized by an exponentially decaying autocorrelation function. As a result, they give rise to a Hurst parameter estimate of $\hat{H} = .50$, producing variance-time curves, R/S plots, and frequency domain behavior strongly disagreeing with the self-similar behavior of actual revocation (see Section 3.3). In this section, we emphasize direct implications of the self-similar nature of the revocation data in the performance of the revocation service.

4.1 Impact on the revocation mechanism

As we mentioned before, traditional mechanisms made assumptions about the revocation process to obtain efficient revocation data issuing policies. However, these assumptions neglect the self-similar nature of the revocation data. This has a direct impact due to the "burstiness" of the data and affects the congestion management of the CA/repositories.

To give an idea of the impact of self-similarity, we analyze the work of Cooper in [9] and in [10]. In these works, Cooper analyzed the best way to issue CRLs, segmented CRLs and delta-CRLs in order to decrease the request peak bandwidth. The author assumed that an average of 1,000 certificates are revoked each day and that the CRLs have a fixed validity time. By doing these assumptions, the self-similar behavior of the revocation process is neglected and the results need to be adapted to the reality.

Using the traditional approach, CRLs are published periodically. Under this assumption, CAs expect that consecutive CRLs should have similar size. However, this assumption is proven completely wrong when bursts are present. Thus, consecutive CRLs can differ significantly in the number of revoked certificates they include, and, consequently in their size. Using

the data collected from Verisign[2], we studied how the size of the CRLs varies when CRLs are issued daily. As in [10], we estimate that the size of a CRL is 51 bytes plus 9 bytes for each certificate included on the CRL. If an average of r certificates are revoked each day, certificates are valid for L_c days, and a certificate, at the time of revocation, has an average of $\frac{L_c}{2}$ days until it expires, then the average size of a CRL will be [10]:

$$Size_{CRL} = 51 + 4.5 \cdot r \cdot L_c.$$

We assume that certificates have a lifetime of 365 days [11], therefore we can calculate the daily size of the Verisign CRL for 5 randomly chosen months. We execute the trial several times and check that the same depency is obtained. Figure 6 shows the results in a box-plot. Note that the CRL size has a mean size of around 150 KBytes, but it highly varies due to the revocation bursts. For instance, during March 2008, there were four CRLs that exceed the 300 KBytes. These variations are highly inefficient in terms of bandwidth, as during some days the required bandwidth double the bandwidth needed in previous days. Although this has not become a bottleneck in wired networks, novel scenarios (e.g. Vehicular Networks) cannot afford these variations.

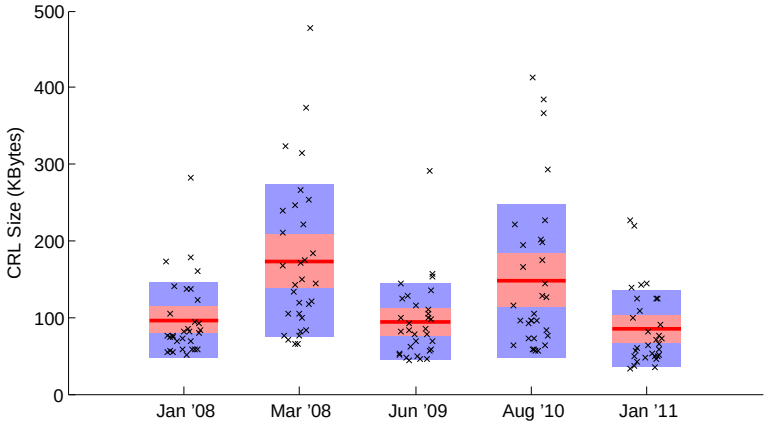


Fig. 6: Estimated daily size of Verisign's CRL.

[2] Note that we use the data from VeriSign to provide a case study of variance in size of the CRLs. The same variance pattern applies to the other CAs, though it is not shown in this article.

However, the self-similarity not only affects traditional CRL issuance, but also its variants that aim to be bandwidth efficient such as delta-CRL. From [10], the bandwidth for a delta-CRL system can be computed as:

$$B = \frac{Nve^{-vt}((51 + 4.5rL_c)e^{-(w+\frac{l}{O}-l)v} + (51 + 9rw))}{(O-1)1 - e^{vl/O} + 1}, \quad (1)$$

where N is the number of valid certificates, v is the validation rate, l is the amount of time that a delta-CRL is valid, L_c is the certificate lifetime, r is the number of certificates revoked per day, w is the window size of the delta-CRL and O is the number of delta-CRLs that are valid at any given time.

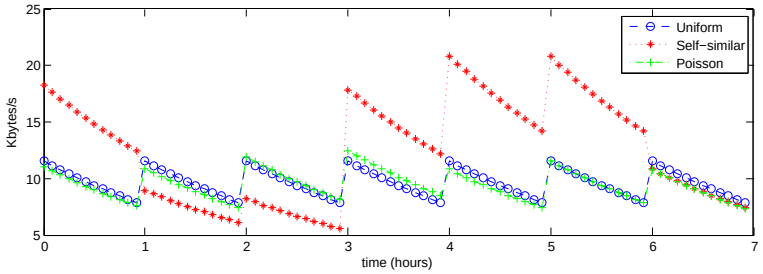


Fig. 7: Delta-CRL BW consumption.

Using the bandwidth as a comparison metric, we can evaluate the impact of the self-similarity. Figure 7 shows the bandwidth necessary to download the revocation data using a sliding window delta-CRL scheme. We have assumed that there are 300,000 relying parties (N) each validating an average of 10 certificates per day (v); delta-CRLs are issued once an hour, are valid for 4 hours (O), and have a window size of 9 hours (w). We have also assumed that an average of 10 certificates are revoked each day (r) and that certificates are valid for 365 days (L_c). Note that depending on the distribution of the revocation process, the required bandwidth presents significant variations. We change the number of certificates revoked per day (r) according to three different distributions (i.e. uniform, Poisson and self-similar) and evaluate the required bandwidth of a delta-CRL system using Eq. (1). Uniform and Poisson distributions present a similar behavior. On the opposite, a self-similar process makes the delta-CRL's size to vary. Thus, the optimal window to issue delta-CRLs should be calculated taking into account the bursty

pattern of the self-similar process. If this pattern is neglected, the peak bandwidth will vary with each delta-CRL issuance making the revocation service bandwidth-inefficient. When with a Poisson or uniform process the maximum peak bandwidth is of ∼12Kb/s, a burst of revocation events causes that some delta-CRL issuance require more than ∼20Kb/s. Therefore, ignoring the self-similar pattern of the revocation process leads to inaccurate network planning.

CRL releasing strategies might be optimized considering the effect of self-similarity. Periodic updates might create bottlenecks at the repositories when all users request new information at the same time. On the other hand, online checking mechanisms such as OCSP, could be computationally overloaded during bursty periods. Such mechanisms that base their efficiency on using pre-signed responses have not been conceived to work under bursty patterns. Therefore, further analysis should be conducted to establish pre-signing policies under bursty revocation periods.

5 Related Work

Most of previous studies fail to capture the characteristics of real-world revocation data; instead, they focus on theoretical aspects of certificate revocation including the model of revocation [9], the revocation cause [12], and the cost of issuing revocation information [13]. Thus, these theoretical models are not able to capture the actual pattern of the revocation data. Most recently, the statistical properties of real revocation data have been studied [14–16]. Nevertheless, the bursty pattern of the revocation process is neglected.

Regarding the traditional way of issuing CRLs, X.509 [17] defines one method to release CRLs. This method involves each CA periodically issuing CRLs. Using this method, the number of revoked certificates contained in each CRL varies significantly. Thus, each CRL has a different size, and the issuance of the CRLs results bandwidth inefficient. Authors in [15] already acknowledged the inefficiencies of the traditional method, and proposed releasing CRLs based on a set of economic costs. However, they assumed a Poisson process when characterizing the number of new certificate revocations, i.e., they neglected the burst pattern. Thus, the resulting CRL releasing policies could be improved by taking into account the self-similarity of the revocation process. Similarly, authors in [18] collected empirical data about the reasons and frequency of user terminations that require certificate revocations, and then model the consequences for certificate revocation. They investigate how to reduce the

cost of certificate revocation by reducing the number of revoked certificates and bandwidth consumption in order to achieve better scalability.

In the same manner, authors in [14] carried out a thorough empirical analysis of the revocation data not only taking into account the number of revoked certificates, but also other factors such as geographical regions and revocation causes. They also conclude that their collected CRLs exhibit exponential distribution patterns. Though they acknowledge the existence of revocation bursts, they do not capture this behavior. On the other hand, authors in [16] suggest a functional form for the probability density function of certificate revocation requests. They choose an exponential distribution function because it adequately approximates the data they collected from a single CA. Based on this assumption, they provide an economic model based on which a CA can choose what they state to be the optimal CRL release interval. However, they do not take into account the self-similar behavior of the revocation data.

6 Conclusions

Current simulation studies for performance evaluation and revocation data release strategies most commonly assume that the temporal distribution of revocation events follows a Poisson distribution. In this paper, we questioned the assumption of Poisson distribution. Our analysis of the revocation data contained in different CRLs provides significant evidence that the real revocation events follow a self-similar distribution. In particular, our analysis showed burstiness at all time-scales, confirming scale-invariance of distribution. We also estimated and showed that Hurst parameter for the daily number of revoked certificates is above 0.5, proving the self-similarity and Long Range Dependence formally.

We then turned our attention to understanding its consequences on the performance of the revocation services. We showed that traditional revocation mechanisms, such as CRLs or delta-CRLs, do not take into account the bursty pattern of the revocation events when establishing the issuing strategies. These bursts increase the maximum peak bandwidth required to provide the revocation data timely. Thus, self-similarity has a profound effect on the engineering of traditional mechanisms and should be taking into account when designing new revocation protocols.

References

1. Walter Willinger, Vern Paxson, and Murad S. Taqqu. *Self-similarity and heavy tails: structural modeling of network traffic*, pages 27–53. 1998.

2. J. Beran. *Statistics for Long-Memory Processes*. Monographs on Statistics and Applied Probability. Chapman & Hall, 1994.
3. Murad S. Taqqu, Vadim Teverovsky, and Walter Willinger. Estimators for long-range dependence: An empirical study. *Fractals*, 3:785–798, 1995.
4. C K Peng, S Havlin, H E Stanley, and A L Goldberger. Quantification of scaling exponents and crossover phenomena in nonstationary heartbeat time series. *Chaos Woodbury Ny*, 5(1):82–87, 1995.
5. Netcraft. Market share of certification authorities, 2009. https://ssl.netcraft.com/ssl-sample-report/CMatch/certs Accessed on 05/2011.
6. Gaurav Jain. Certificate revocation: A survey. http://csrc.nist.gov/pki/welcome.html Accessed on 05/2011.
7. Thomas Karagiannis, Michalis Faloutsos, and Rudolff H. Riedi. Long-range dependence: now you see it, now you don't. In *in Proc. GLOBECOM '02*, pages 2165–2169, 2002.
8. Will E. Leland, Murad S. Taqqu, Walter Willinger, and Daniel V. Wilson. On the self-similar nature of ethernet traffic (extended version). *IEEE/ACM Trans. Netw.*, 2(1):1–15, feb 1994.
9. D.A. Cooper. A model of certificate revocation. In *Fifteenth Annual Computer Security Applications Conference*, pages 256–264, 1999.
10. D.A. Cooper. A more efficient use of Delta-CRLs. In *2000 IEEE Symposium on Security and Privacy. Computer Security Division of NIST*, pages 190–202, 2000.
11. Technological infrastructure for pki and digital certification. *Computer Communications*, 24(14):1460 – 1471, 2001.
12. B. Fox and B. LaMacchia. Certificate Revocation: Mechanics and Meaning. In *International Conference on Financial Cryptography (FC98)*, volume 1465, pages 158–164, February 1998.
13. M. Naor and K. Nissim. Certificate Revocation and Certificate Update. *IEEE Journal on Selected Areas in Communications*, 18(4):561–560, 2000.
14. Daryl Walleck, Yingjiu Li, and Shouhuai Xu. Empirical analysis of certificate revocation lists. In *Proceeedings of the 22nd annual IFIP WG 11.3 working conference on Data and Applications Security*, pages 159–174, 2008.
15. Chengyu Ma, Nan Hu, and Yingjiu Li. On the release of CRLs in public key infrastructure. In *Proceedings of the 15th conference on USENIX Security Symposium*, volume 15, pages 17–28, 2006.
16. Nan Hu, Giri K. Tayi, Chengyu Ma, and Yingjiu Li. Certificate revocation release policies. *Journal of Computer Security*, 17:127–157, April 2009.
17. ITU/ISO Recommendation. X.509 Information Technology Open Systems Interconnection - The Directory: Autentication Frameworks, 2000. Technical Corrigendum.
18. Mona Ofigsbø, Stig Mjølsnes, Poul Heegaard, and Leif Nilsen. Reducing the cost of certificate revocation: A case study. In *Public Key Infrastructures, Services and Applications*, volume 6391 of *Lecture Notes in Computer Science*, pages 51–66. 2010.

PKIX Certificate Status in Hybrid MANETs

Jose L. Muñoz, Oscar Esparza, Carlos Gañán, Javier Parra-Arnau

Universitat Politècnica de Catalunya (Departament Enginyeria Telemàtica)**
1-3 Jordi Girona, C3 08034 Barcelona (Spain)
{jose.munoz,oscar.esparza,carlos.ganan,javier.parra}@entel.upc.es

Abstract. Certificate status validation is a hard problem in general but it is particularly complex in Mobile Ad-hoc Networks (MANETs) because we require solutions to manage both the lack of fixed infrastructure inside the MANET and the possible absence of connectivity to trusted authorities when the certification validation has to be performed. In this sense, certificate acquisition is usually assumed as an initialization phase. However, certificate validation is a critical operation since the node needs to check the validity of certificates in real-time, that is, when a particular certificate is going to be used. In such MANET environments, it may happen that the node is placed in a part of the network that is disconnected from the source of status data at the moment the status checking is required. Proposals in the literature suggest the use of caching mechanisms so that the node itself or a neighbour node has some status checking material (typically on-line status responses or lists of revoked certificates). However, to the best of our knowledge the only criterion to evaluate the cached (obsolete) material is the time. In this paper, we analyse how to deploy a certificate status checking PKI service for hybrid MANET and we propose a new criterion based on risk to evaluate cached status data that is much more appropriate and absolute than time because it takes into account the revocation process.

Keywords: Certification, Public Key Infrastructure, Revocation, Hybrid MANET, Risk.

1 Introduction

MANETs (Mobile Ad-hoc Networks) are cooperative networks that allow wireless nodes to establish spontaneous communications. As stated in [1], such networks are envisioned to have dynamic, sometimes rapidly-changing, random, multi-hop topologies which are likely composed of relatively bandwidth constrained wireless links. MANETs may operate in

** This work is funded by the Spanish Ministry of Science and Education under the projects CONSOLIDER-ARES (CSD2007-00004), SECCONET (TSI2005-07293-C02-01), ITACA (TSI2007-65393-C02-02), P2PSEC (TEC2008-06663-C03-01) and, by the Government of Catalonia under grant 2005 SGR 01015 to consolidated research groups.

isolation (stand-alone), or they may have gateways to fixed networks. In this last case, the MANET is called "hybrid". Hybrid MANETs are expected to be deployed as an extension to the traditional infrastructure networks. Also notice that the hybrid behaviour can be temporary due to the situation in which an ad-hoc network may be sometimes stand-alone and sometimes connected to the Internet e.g. a subway network in which a MANET user is connected to the Internet while being at the station and disconnected while traveling. The Hybrid MANET scenario is the one considered in this paper.

On the other hand, trust and security are basic requirements to support business applications in this scenario. The public key scheme is the preferred underlying mechanism to provide security services. In a public key scheme, each participant has two keys: a public key (i.e. known by everybody) and a private key (i.e. secret). The announcement of the public key is performed using a signed document called Public Key Certificate (PKC) or simply "certificate" that binds the participant with her public key. The entity that signs the certificate is called "certificate issuer" or "Certification Authority" (CA). In the literature, there are several ways of managing security and trust in MANETs based on public key cryptography. These approaches basically differ in the degree of decentralization of the mechanisms deployed for issuing, publishing and revoking the certificates (these approaches are reviewed in further detail in the next section).

In decentralized architectures such as [2] and [3] the nodes inside the ad-hoc network participate in the certification process. On the other hand, in the centralized architecture the certification process is fully controlled by an external CA that is a Trusted Third Party (TTP). In this case the CA digitally signs certificates, ensuring that a particular public key belongs to a certain user and the overall certification process is performed according to a standard and publicly available policy. Each scheme has its application scenario: decentralized approaches are suitable for autonomous MANETs or hybrid MANETs that do not require a centralized enforced certification mechanism while the centralized approach is suitable for hybrid MANETs in which inter-operability with currently deployed centralized public key infrastructures (PKIs) is required.

The problem of using a centralized approach is that current PKIs are designed for wired and well-connected networks, so adopting PKIs for hybrid MANETs is not an easy task. Mobile users are expected to move across different networks. When the user is in a network with connection to the PKI, she can use all the PKI services such as get a certificate,

launch a status query, etc. However, users may be disconnected from the PKI when they require a real-time PKI service. In this sense, the certificate status checking is a critical service because applications must decide, at the time of usage, whether a certificate is acceptable or not to perform an action. Proposals in the literature suggest the use of caching mechanisms to let the node itself or a neighbour node to store status checking material (typically on-line status responses or lists of revoked certificates). However, to the best of our knowledge the only criterion to evaluate the cached (obsolete) material is the time. In this paper we propose and formulate a new criterion based on risk to evaluate cached status checking data that is much more appropriate and absolute than time because it takes into account the revocation process. The rest of the paper is organized as follows: Section 2 presents an analysis of the main certification approaches for MANET. Section 3 discusses the main issues that have to be solved in order to adapt current PKI status checking mechanisms to MANET. In Section 4, we present our proposal to evaluate cached status data and, finally, we conclude in Section 5.

2 Certificate Management schemes for MANET

In general, certificate management schemes can be classified as:

- Decentralized. The nodes of the MANET participate either fully or partially in the certification process (see Figure 1.b).
- Centralized. Authorities outside the MANET control the certification process according to a global policy (see Figure 1.a).

In the fully decentralized PKI schemes for MANET, like Capkun et al. [3, 4], the nodes of the MANET themselves issue, publish and revoke the certificates. The certificate management is autonomous and self-organized because there is no need for any trusted authority or fixed server and all the nodes have the same role. In this system, like in PGP (Pretty Good Privacy) [5], each user is her own issuer. Certificates are stored and distributed by the nodes in a fully self-organized manner. Each certificate is issued with a limited validity period and it contains its issuing and expiration times. Before a certificate expires, the owner can issue an updated version of the certificate, which contains an extended expiration time. Authors call this updated version the certificate update. Each node periodically issues certificate updates, as long as the owner considers that the user-key bindings contained in the certificate are correct. Trust is achieved via chains of certificates. The nodes build trust paths certifying from one

node to another, as in a friendship circle, forming an authentication ring to achieve the trust relationships with other nodes of the MANET. A decentralized trust management model for pervasive computing environments is presented in [6], where authors overcome the challenges posed by dynamic open environments, making use of the autonomy and cooperative behaviour of the entities.

Another group of public key schemes for MANET is based on threshold cryptography [2]. The idea behind these schemes is to distribute certification duties amongst network nodes. A (k, n) threshold scheme allows the signing private key to be split into n shares such that any k nodes could combine and recover the signing key for a certain threshold $k < n$, whereas $k-1$ or fewer nodes are unable to do so. In this manner, the signing key can be partitioned into n shares and distributed to n nodes using the previous cryptographic technique. For instance, any k of n nodes could then collaborate to sign and issue valid digital certificates or issue status data; whereas a coalition of $k-1$ or fewer nodes would not be able to do so. Notice that this scheme is partially decentralized because it requires an initialization phase in which a centralized authority assigns the role to the n nodes that will act as servers for certificate management. Partially decentralized schemes were first proposed by Zhou and Haas in [7]. This work inspired a practical system called COCA [8] in which a threshold cryptography scheme is implemented for infrastructure-based networks. On the other hand, another system called MOCA [9] extends this idea to ad-hoc networks. In this scheme security is improved by selecting powerful nodes as Certificate Authority servers.

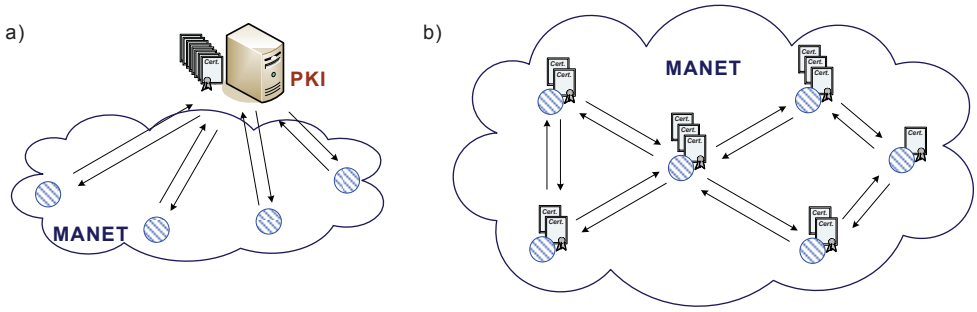


Fig. 1. Centralized and decentralized schemes

Finally, an external public key infrastructure can also be used for the hybrid scenario. In this case, centralized trusted authorities issue, publish and distribute the status (valid/revoked) of certificates according to a well-defined standard methodology. In the Internet, the PKIX [10] is the currently working public key infrastructure. However, PKIX is mostly designed for wired and well-connected networks and adapting the PKIX to the hybrid scenario is a challenging task because MANET nodes are expected to move across different networks, sometimes with on-line connection to the PKIX services and sometimes not. When the user is in a network with connection to the PKI, she can use all the PKI services such as getting a certificate, launching a status query, etc. However, users may be disconnected from the PKIX when they require real-time PKIX services. We discuss the problem of adapting PKI to MANET in more detail in the next section.

3 Adapting PKIX to MANET

The local validity of the certificates in the decentralized approaches may restrict their usability in the hybrid scenario. In this sense, the PKIX approach is suitable for hybrid MANETs that require support for mobility maintaining a centralized enforced certification mechanism and also interoperability with currently deployed PKIs. However, the original design of the PKIX assumes that the user can access at any time to the entities of the infrastructure which is true for wired well-connected networks but not for our scenario.

The first problem that we have to face is the certificate acquisition. A permanent connection of the client to the infrastructure cannot be assumed so the solution is to choose relatively long validity periods for the certificates. The idea is that the user has to pass an initial certification process before she can start operating in the MANET. Once the user has its credential, she can operate in the hybrid scenario without further interaction with the PKI (at least interaction is not required for a quite long time). This way of issuing the certificates can be assumed as an initialization phase equivalent to the initialization phase of the partially decentralized scheme in which the shares are delivered.

On the other hand, a certificate might be revoked (invalidated) prior to its expiration. Among other causes, a certificate may be revoked because of the loss or compromise of the associated private key, in response to a change in the owner's access rights, a change in the relationship with the issuer or as a precaution against cryptanalysis. The revocation policies

determine how the status of the certificates is distributed to the end users. So the PKI is responsible for the certificates not only at the issuing time but also during all the certificate's life-time.

The problem is that PKIX explicit revocation systems were designed for wired and well-connected networks in which repositories and responders have a well-known network address and are always available to users. However, MANETs are dynamic environments in which network topology changes randomly and in which mobile users continuously join and leave the network. Therefore, new mechanisms are necessary to distribute explicit status data in MANETs. Proposals in the literature suggest the use of caching mechanisms to address these problems.

Caching schemes allow to manage arbitrary disconnections between the users and the sources of the status data service. Disconnections are alleviated by storing copies of status data (lists of revoked certificates or on-line responses) in the nodes of the ad-hoc network. These copies are obtained when connection to the infrastructure is available. In general, an ad-hoc caching scheme for any service has four different kinds of nodes [11]: server-nodes, client-nodes, caching-nodes and intermediate-nodes (see Figure 2). For the status checking service:

- *Server-nodes.* These nodes have "always updated data" to offer the status checking service. The server-node has a permanent connection to the certification infrastructure in order to have always fresh status information. Typically, a server-node is an Access Point connected to both to a MANET and to the fixed network.
- *Client-nodes.* These nodes require the status checking service. A *service discovery* mechanism has to be provided to the client so that she can find a node in the network that provides the service.
- *Caching-nodes.* These nodes have cached data and therefore they may also provide the status checking service. A client-node in the absence of connectivity to a server-node or because of performance issues can connect with a close caching-node to obtain the service with cached status data (perhaps quite obsolete data).
- *Intermediate-nodes.* These nodes forward the packets among client and server nodes. They may also store the path to a service provider (whether a server-node or a caching-node) together with service parameters such as data size, the service expected Time-To-Live (TTL), number of hops to reach the provider etc.

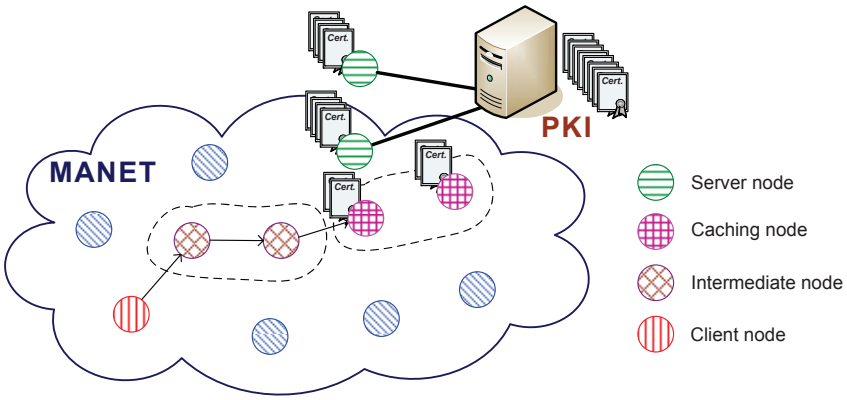


Fig. 2. Four different kinds of nodes in caching schemes.

In the literature we can find some proposals that apply the previous ideas to adapt the PKI status checking standards CRL [12, 13] and OCSP [14] to the MANET. A CRL is a black list with the identifiers of revoked certificates. The integrity and authenticity of the CRL is provided by an appended digital signature. On the other hand, OCSP is a protocol to make the status of certificates available through a request/response mechanism. The OCSP server is called responder and provides signed responses to clients. Next, we give our point of view about this adaptation and we briefly review some remarkable works about this in the literature.

In the case of CRL, server-nodes are nodes that can maintain a stable connection to PKI repositories in order to get the most updated CRL. A caching-node is a node that is willing to collaborate in the certificate status checking service and that has enough cache capacity to store a CRL copy. The caching-node responds to the status requests of client-nodes in the MANET. Notice that a client-node that acquires a valid CRL copy can become a new caching-node. Furthermore, a caching-node that moves to another MANET can collaborate in the new network to provide the service. In this sense, user's mobility helps the status checking service. In [15], the authors investigate the feasibility of using flooding to distribute CRL information in MANETs by simulation. They conclude that the two major factors for flooding to work smoothly are the number of nodes and the communication range. In [16] a MANET cooperative mechanism for certificate validation is presented in order to overcome both the lack of infrastructure and the limited capabilities of the nodes. This solution is

based on an extended-CRL where the repositories can build an efficient structure through an authenticated hash tree.

Regarding OCSP, server-nodes are responders. We can consider that there are only responders placed in the PKI (fixed-responders) or we can consider the possibility of having responders implemented in a mobile node that can be part of a MANET (mobile-responders). Despite this possibility, we discourage the use of mobile-responders because they are server-nodes and as such they are supposed to have updated status data. A server-node for certificate status checking must have connectivity with PKI repositories or fixed-responders to get updated status data but this connectivity is not always guaranteed in a MANET. On the other hand, a responder is a trusted authority so it has a private key that has to protect against intruders. In our view, it makes no sense having a server-node that is exposed to attacks and that may not have useful data. Furthermore, in general, increasing the number of trusted authorities in a system is not desirable, the less number of trusted authorities, the less is the probability of having a private key compromised. Besides, if mobile-responders are used, it is necessary to define a mechanism to trust them which is not trivial. With respect caching-nodes, they store OCSP responses issued by server-nodes and distribute them to client-nodes when they detect a request that fulfils freshness requirements. In [17, 18], there is a complete proposal called ADOPT (Ad-hoc Distributed OCSP for Trust) that describes a caching scheme for OCSP in MANET.

4 Evaluation of cached status data based on Risk

As explained in the previous section, caching and discovery mechanisms are necessary to manage the situation in which a user is not able to reach a PKI status data server. When a disconnection happens, the client-node uses service discovery to find a caching node. Then, the node obtains a cached version of available status data and finally, the node decides what to do with the data. In this sense, the CA issues status data bounded by two time-stamps:

- *thisUpdate.* Instant at which status data have been issued.
- *nextUpdate.* Instant at which updated status data are expected to be issued.

Let us define T_s as the issuing interval of status data (1).

$$T_s = nextUpdate - thisUpdate \tag{1}$$

As data in status responses are time-stamped, users can get an idea about how fresh is the status of a certificate by looking at the *thisUpdate* parameter of the response and, finally a user can take a decision about whether operate or not with a certain certificate. According to [19] the time is the only criterion to help the user to take this decision and to the best of our knowledge this is the only criterion proposed in the literature. However, this is a poor criterion that can be enhanced. In this section, we propose other parameter rather than time to take this decision.

First of all, let us illustrate why time is a poor parameter for our purposes. For instance, consider a status response issued a couple of hours ago. We may wonder: *is it fresh or not?* The answer is obviously that *"it depends"*. Two hours may not be considered a long time if there are a couple of revoked certificates every month but this period can be considered quite long if there are two new revoked certificates per hour. Moreover, a scenario with millions of issued non-expired certificates is not the same as a scenario that has hundreds of certificates. In the former, a couple of new revoked certificates is not so relevant while in the latter a couple of new revocations is quite important. As a conclusion, we need a parameter that considers all these aspects. For this purpose, we define a risk function that aids the user to decide whether to trust or not a certificate. We formally define the function *risk* ($r(t)$) as the *probability of considering a certificate as a valid one when the real status known by the PKI is revoked at time t*.

To find an analytical expression for the risk function we first need to analyse the certificate issuing process. Certificates are issued with a validity period T_c. Obviously $T_c >> T_s$, for instance T_c can be a year while the period of status data issuing can be an hour. The number of *non-expired certificates* ($N(t)$) -including revoked and non revoked certificates- is a stochastic process whose mean value at instant t depends on the certificate issue and certificate expiration processes. It is assumed that the elapsed time since issuing until expiration (T_c) is a constant value for all certificates. Therefore, the expiration process is the same as the issuance process elapsed T_c time units. This process is defined by the certificate issue rate λ_c, which matches with the certificate expiration rate. Hence the mean value of *non-expired certificates* in steady state is the mean quantity of issued certificates before the expiration process begins.

$$E[N(t)] = N = \lambda_C T_C, \quad t > T_C \tag{2}$$

On the other hand, there is a group of *revoked non-expired certificates*, that is to say, certificates that have a valid validity period but that have

been revoked prior to the expiration date and, therefore they are included in the black list. The subset of *revoked non-expired certificates* is included in the set of *non-expired certificates* and the cardinality of that set, $R(t)$, is a stochastic process that it is typically modelled [20] as a fraction or percentage ($p(t)$) of the non-expired certificates (3).

$$R(t) = p(t)N(t) \quad with\ p(t) \leq 1 \tag{3}$$

Assuming that both processes are independent and using expected values:

$$E[R(t)] = E[p(t)]E[N(t)] \tag{4}$$

$$R = pN \tag{5}$$

We further model the expected percentage of revoked certificates as directly proportional to the certification time T_c (6).

$$p = p'T_c \tag{6}$$

This means that larger certification periods will imply more percentage of revoked certificates. On the other hand, smaller certification periods mean less probability of a certificate being revoked during its life-time and therefore low percentage of revoked certificates. Then, the mean value of the *revoked non-expired certificates* can be expressed as:

$$R = p'\lambda_c T_c^2 \tag{7}$$

We have modelled the issuing and revoking processes of the overall system. However, our goal is to model the risk from the point of view of the user, that is to say, we want to find the probability of considering a certificate as a valid one when the real status known by the PKI is revoked.

Let us assume, without loss of generality, that at instant $t_0 = thisUpdate$ a user gets the current black list of revoked certificates from the PKI. Using this list, the user can split the set of *non-expired certificates* into *revoked certificates* and *not revoked certificates*.

Next, we need to define the subset of *operative certificates* as the group of *non-expired certificates* for which the last status known by a user is *not revoked*. Notice that the PKI may know that a certificate considered operative by a user is in fact revoked. However, due to the MANET conditions it is impossible to communicate this situation to the user.

Now, let us assume that the user is not able to connect to the infrastructure any more. As time goes by the set of *operative certificates* will include revoked certificates and the user will need to take decisions about using an operative certificate assuming a certain risk. The *risk function* $r(t)$ can be evaluated as the ratio between the number of *unknown revoked operative certificates* ($R'(t)$) and the number of *operative certificates* ($N'(t)$) as shown in equation (8).

$$r(t) = \frac{E[R'(t)]}{E[N'(t)]} \tag{8}$$

$N'(t)$ (*number of operative certificates*) can be defined as the number of certificates that were not included in the last black list obtained by the user (were not revoked before t_0) and that they have not expired at t. Included in the set of *operative certificates* there is the subset of *unknown revoked operative certificates*. The cardinality of this subset $R'(t)$ is the number of *operative certificates* that are revoked at instant t, that is, they are revoked but this fact is unknown to the user.

At $t_0 = thisUpdate$ the set of *operative certificates* is the same that the set of *not revoked certificates* and, since the user has the same information that the PKI so there is no risk ($r(t_0) = 0$). Besides

$$E[N'(t_0)] = (1-p)N \tag{9}$$

$$E[R'(t_0)] = 0 \tag{10}$$

At the instant $t_0 + T_C$ all the certificates included in the black list will be expired. This means that all *non expired certificates* will be *operative*, and any revoked certificate will be unknown to the user. The *risk* at this moment can be expressed as (11).

$$r(t_0 + T_C) = \frac{E[R'(t_0 + T_C)]}{E[N'(t_0 + T_C)]} = \frac{E[R(t_0)]}{E[N(t_0)]} = p \tag{11}$$

To evaluate the function risk between t_0 and $t_0 + T_C$ we have to observe the processes $N'(t)$ and $R'(t)$ in this interval. After t_0 the variation of the number of *operative certificates* ($N'(t)$) depends on these factors:

- Increases because of the new issues.
- Decreases because of the expiration of operative certificates issued before instant t_0 (the certificates issued later do not expire in the considered interval).

The issuance rate is λ_c that is the same as the expiration rate. But notice that not all expirations concern to *operative certificates*. A fraction p of the expirations corresponds to *revoked non expired certificates*, and the other fraction $1 - p$ corresponds to *operative certificates*. Then the expiration rate of *operative certificates* is $(1 - p)\lambda_c$ (see Figure 3).

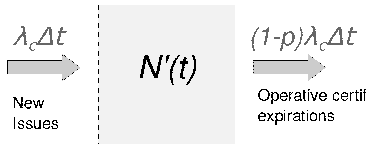


Fig. 3. Evolution of operative certificates

Considering the evolution of the set of *operative certificates* we can evaluate its expected cardinal (12).

$$E[N'(t)] = E[N'(t_0)] + \lambda_C(t - t_0) - (1 - p)\lambda_C(t - t_0) \tag{12}$$

Using (9) we obtain.

$$E[N'(t)] = (1 - p)N + p\lambda_C(t - t_0) \tag{13}$$

Finally, we need an expression for the set of *revoked operative certificates*. This set is the intersection of the set of *operative certificates* and the set of revoked certificates as shown in the Figure 4.

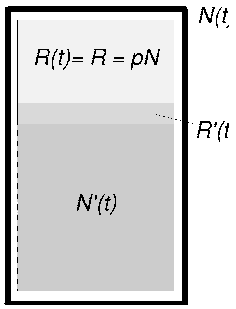


Fig. 4. Sets of certificates

Hence we can express the cardinality of these sets using the following expression.

$$N(t) = R(t) + N'(t) - R'(t) \tag{14}$$

Therefore,

$$R'(t) = R(t) + N'(t) - N(t) \tag{15}$$

We obtain the expected value of the number of revoked operative certificates using (15), (2), (5) and (13).

$$E[R'(t)] = p\lambda_C(t - t_0) \tag{16}$$

To obtain the *risk* function we use the expressions (13), (16) and the expression of its definition (8).

$$r(t) = \frac{p(t - t_0)}{(1 - p)T_c + p(t - t_0)} \tag{17}$$

The previous expression is valid for instants of time $t \ \epsilon \ t_0 \leq t \leq t_0 + T_c$ and fulfils with the expected results of expressions (10) and (11). Notice that the risk function allows a user to compute the probability of considering a non-expired certificate as non-revoked when the real status known by the PKI is revoked.

On the other hand, it is remarkable that unlike time which is a relative parameter, the risk function gives the user an absolute parameter to aid her taking the decision of trusting or not a particular certificate. This decision must be taken when the user is disconnected from the infrastructure and therefore it is taking into consideration cached (obsolete) status data.

Finally, the risk function should be used as follows:

- In first place, the CA signs the status data with the two standard time-stamps (*thisUpdate* and *nextUpdate*) but it also adds the current parameter p. The CA can calculate this parameter because it knows the current number of issued non-expired certificates and the current number of non-expired revoked certificates.
- When the user has to evaluate status data, she knows T_c as this is the certification period included in her certificate.
- Then, the user obtains p from the status data.

– Next, the user can compute the risk at current time t by replacing t_0 with *thisUpdate* in the risk function.
– Finally, the user can take a decision about a target certificate with the risk value computed.

5 Conclusions

Decentralized certification architectures for MANET such as self-organized PKIs and PKIs based on threshold cryptography generally provide certificate validation mechanisms inside the MANET. However, local validity of the certificates and inter-operability with currently deployed PKIs may restrict their usability in an hybrid MANET scenario. If a centralized certification infrastructure such as PKIX is used, then certificate validation becomes one of the main problems. This is because users need to ensure at the time of usage that the certificate they are relying upon has not been revoked but at the same time trusted servers of PKIX may be unavailable. Besides, standard status checking mechanisms of the fixed network are not directly usable because they are designed for always connected users.

In this sense, caching schemes allow to manage arbitrary disconnections between the users and the sources of the status data service. Disconnections are alleviated by storing copies of status data (lists of revoked certificates or on-line responses) in the nodes of the ad-hoc network. These copies are obtained when connection to the infrastructure is available. On the other hand, a service discovery mechanism is necessary to find the nodes that have cached material. In this paper, we have reviewed and analysed all these issues for adapting the standard PKIX status checking mechanisms to hybrid MANET.

Despite the caching scheme allows the users to obtain status data during disconnections, the cached status data is likely to be outdated. When using cached status data a node could operate with a revoked certificate considering it is a valid one. In this paper, we have presented a novel scheme which provides users within the MANET with an absolute criterion to determine whether to use or not a target certificate when updated status data is not available. By taking into account information about the revocation process, users can calculate a *risk* function in order to estimate whether a certificate has been revoked while there is no connection to a status checking server. Finally, it is also worth to mention that this new criterion can be applied to other networks than hybrid MANETs if these networks are based on an off-line explicit revocation scheme.

Abbreviations

ADOPT Ad-hoc Distributed OCSP for Trust.
CA Certification Authority.
COCA Cornell On-line Certification Authority.
CRL Certificate Revocation List.
MANET Mobile Ad-hoc Network.
MOCA Mobile Certificate Authority.
OCSP On-line Certificate Status Protocol.
PGP Pretty Good Privacy.
PKI Public Key Infrastructure.
PKIX Public Key Infrastructure (X.509).
TTL Time-To-Live.
TTP Trusted Third Party.

References

1. S. Corson and J. Macker. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. RFC 2501 (Informational), January 1999.
2. Y. Desmedt and Y. Frankel. Threshold cryptosystems. in advances in cryptology—crypto'89. In *the Ninth Annual International Cryptology Conference*, volume 435 of *LNCS*, pages 307–315. Springer-Verlag, 1989.
3. S. Capkun, L. Buttyan, and J.P. Hubaux. Self-organized public-key management for mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 2003.
4. J-P. Hubaux, L. Buttyan, and S. Capkun. The quest for security in mobile ad hoc networks. In *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC'01)*, 2001.
5. J. Zsako. PGP Authentication for RIPE Database Updates. RFC 2726 (Proposed Standard), December 1999.
6. F. Almenárez, A. Marín, C. Campo, and C. García. Managing ad-hoc trust relationships in pervasive environments. In *Proceedings of the Workshop on Security and Privacy in Pervasive Computing SPPC*, 2004.
7. L. Zhou and Z.J. Haas. Securing ad hoc networks. *IEEE Networks*, 13(6):24–30, 1999.
8. L. Zhou, F.B. Schneider, and R.V. Renesse. Coca: A secure distributed on-line certification authority. *ACM Transactions on Computer Systems*, 20(4):329–368, 2002.
9. S. Yi and R. Kravets. Moca: Mobile certificate authority for wireless ad hoc networks. In *Proceedings of the 10th IEEE International Conference on Network Protocols (ICNP'O2)*, 2002.
10. Pkix chapter of the ietf. www.ietf.org/html.charters/pkix-charter.html.
11. L. Yin and G. Cao. Supporting cooperative caching in ad hoc networks. *IEEE Transactions on Mobile Computing*, 5(1):77–89, 2006.
12. R. Housley, W. Ford, W. Polk, and D. Solo. Internet X.509 Public Key Infrastructure Certificate and CRL Profile. RFC 2459 (Proposed Standard), January 1999. Obsoleted by RFC 3280.

13. S. Tuecke, V. Welch, D. Engert, L. Pearlman, and M. Thompson. Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile. RFC 3820 (Proposed Standard), June 2004.
14. M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP. RFC 2560 (Proposed Standard), June 1999.
15. H. W. Go, P. Y. Chan, Y. Dong, A. F. Sui, S. M. Yiu, Lucas C. K. Hui, and Victor O. K. Li. Performance evaluation on crl distribution using flooding in mobile ad hoc networks (manets). In *ACM Southeast Regional Conference archive. Proceedings of the 43rd annual southeast regional conference*, volume 2, pages 75–80, Kennesaw, Georgia, 2005.
16. J. Forné, J. L. Muñoz, O. Esparza, and F. Hinarejos. Certificate status validation in mobile ad hoc networks. *IEEE Wireless Communications*, 16(11):55–62, 2009.
17. G. F. Marias, K. Papapanagiotou, V. Tsetsos, O. Sekkas, and P. Georgiadis. Integrating a trust framework with a distributed certificate validation scheme for manets. *Wireless Communications and Networking*, 1155(10):1–18, 2006.
18. 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*, 2006(2):1–18, 2006.
19. A. Deacon and R. Hurst. The Lightweight Online Certificate Status Protocol (OCSP) Profile for High-Volume Environments. RFC 5019 (Proposed Standard), September 2007.
20. A. Arnes. Public key certificate revocation schemes, February 2000. Queen's University. Ontario, Canada. Master Thesis.

References

[1] WhichSSL, "SSL Market Share," 2010, [Online] Available: http://www.whichssl.com/ssl-market-share.html.

[2] "Car2car Communication Consortium," [Online] Available: http://www.car-to-car.org.

[3] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Secur.*, vol. 15, pp. 39–68, January 2007.

[4] A. S. for Testing and M. (ASTM), "Standard specification for telecommunications and information exchange between roadside and vehicle systems-5ghz band dedicated short range communications (DSRC) medium access control (MAC) and physical layer (PHY) specifications," ASTM International, Technical Report ASTM E2213 - 03(2010), 2010.

[5] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, ser. SASN '05, 2005, pp. 11–21.

[6] J. P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *Security Privacy, IEEE*, vol. 2, no. 3, pp. 49 –55, May 2004.

[7] P. Papadimitratos, L. Buttyan, J. P. Hubaux, F. Kargl, A. Kung, and M. Raya, "Architecture for secure and private vehicular communications," in *Telecommunications, 2007. ITST '07. 7th International Conference on ITS*, Jun. 2007, pp. 1 –6.

[8] "IEEE trial-use standard for wireless access in vehicular environments - security services for applications and management messages," *IEEE Std 1609.2-2006*, pp. 1–105, 2006.

[9] J. Iliadis, D. Spinellis, D. Gritzalis, B. Preneel, and S. Katsikas, "Evaluating certificate status information mechanisms," in *Proceedings of the 7th ACM conference on Computer and communications security*, ser. CCS '00. New York, USA: ACM, 2000, pp. 1–8.

[10] G. F. Marias, K. Papapanagiotou, and P. Georgiadis, "ADOPT. a distributed OCSP for trust establishment in MANETs," in *11th European Wireless Conference 2005*, 2005.

[11] T. P. Hormann, K. Wrona, and S. Holtmanns, "Evaluation of certificate validation mechanisms," *Comput. Commun.*, vol. 29, pp. 291–305, February 2006.

[12] A. Arnes, M. Just, S. J. Knapskog, S. Lloyd, and H. Meijer, "Selecting revocation solutions for PKI," in *NORDSEC '00*, 2000.

[13] D. Walleck, Y. Li, and S. Xu, "Empirical analysis of certificate revocation lists," in *Proceeedings of the 22nd annual IFIP WG 11.3 working conference on Data and Applications Security*, 2008, pp. 159–174.

[14] C. Ma, N. Hu, and Y. Li, "On the release of CRLs in public key infrastructure," in *Proceedings of the 15th conference on USENIX Security Symposium - Volume 15*. Berkeley, CA, USA: USENIX Association, 2006.

[15] N. Hu, G. K. Tayi, C. Ma, and Y. Li, "Certificate revocation release policies," *J. Comput. Secur.*, vol. 17, pp. 127–157, April 2009.

[16] G. E. P. Box and G. Jenkins, *Time Series Analysis, Forecasting and Control*. Holden-Day, Incorporated, 1990.

[17] A. Wasef and X. Shen, "EDR: Efficient Decentralized Revocation Protocol for Vehicular Ad Hoc Networks," *Vehicular Technology, IEEE Transactions on*, vol. 58, no. 9, pp. 5214 –5224, nov. 2009.

[18] K. P. Laberteaux, J. J. Haas, and Y.-C. Hu, "Security certificate revocation list distribution for VANET," in *Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking*, ser. VANET '08, 2008, pp. 88–89.

[19] P. Papadimitratos, G. Mezzour, and J.-P. Hubaux, "Certificate revocation list distribution in vehicular communication systems," in *Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking*, ser. VANET '08, 2008, pp. 86–87.

[20] J. L. Muñoz, O. Esparza, C. Gañán, and J. Parra-Arnau, "PKIX certificate status in hybrid MANETs," in *Information Security Theory and Practice. Smart Devices, Pervasive Systems, and Ubiquitous Networks (WISTP)*, ser. Lecture Notes in Computer Science, vol. 5746. Springer, 2009, pp. 153–166.

[21] B. Fox and B. A. LaMacchia, "Certificate Revocation: Mechanics and Meaning," in *International Conference on Financial Cryptography (FC98)*, no. 1465, Feb. 1998, pp. 158–164.

[22] R. Rivest, "Can we eliminate certificate revocation lists?" in *International Conference on Financial Cryptography*. Springer-Verlag, 1998, pp. 178–183.

[23] P. McDaniel and A. Rubin, "A response to can we eliminate certificate revocation lists," in *International Conference on Financial Cryptography 2000 (FC00)*. Springer-Verlag, Feb. 2000.

[24] J. J. Haas, Y.-C. Hu, and K. P. Laberteaux, "Design and analysis of a lightweight certificate revocation mechanism for VANET," in *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking*, ser. VANET '09. New York, NY, USA: ACM, 2009, pp. 89–98.

[25] B. of Transportation Statistics U.S. Department of Transportation, "Number of U.S. aircraft, vehicles, vessels, and other conveyances," 2009, [Online] Available: http://www.bts.gov/publications/national_transportation_statistics/html/table_01_11.html.

[26] D. N. Cottingham, I. J. Wassell, and R. K. Harle, "Performance of IEEE 802.11a in vehicular contexts," in *Proc. IEEE VTC*. Spring, 2007.

[27] R. Merkle, "A certified digital signature," in *Proceedings of Advances in Cryptology (CRYPTO' 89)*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 1990, vol. 435, pp. 218–238.

[28] "ITU-T X.509 : Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks," International Telecommunication Union, 2005.

[29] L. Lamport, "Password authentication with insecure communication," *Commun. ACM*, vol. 24, pp. 770–772, November 1981.

[30] S.-Y. Wang and C.-L. Chou, "NCTUns tool for wireless vehicular communication network researches," *Simulation Practice and Theory*, vol. 17, pp. 1211–1226, 2009.