Postal Revenue Collection in the Digital Age

Leon A. Pintsov[1] and Scott A. Vanstone[2]

[1] Pitney Bowes Inc., Stamford, Connecticut, USA
pintsov@pb.com
[2] University of Waterloo and Certicom Corp., Ontario, Canada
savansto@uwaterloo.ca

Abstract. In recent years postal revenue collection underwent a major transformation due to widespread transition to digital methods of communication. This transition directly affected not only telecommunications which form an integral part of the postal revenue collection but also, and in a much more profound way, postage evidencing. Traditional postage evidencing remained unchanged for several dozens years until the introduction of digital printing which drastically changed all its security related aspects and considerations. This paper defines conceptual foundations of the postal revenue collection system (which is simultaneously a payment system for mailers), fundamental requirements imposed by the nature of hardcopy-based communication and suggests what the authors believe to be an optimal solution for public key-based postage evidencing founded on elliptic-curve cryptography.

1 Background and Introduction

Payment/revenue collection systems are critical parts of business enterprises. This paper is concerned with postal revenue collection (which is simultaneously a payment system for mailers) from a system viewpoint. Postal revenues amount to well over 150 billion US dollars annually. Economic efficiency of payment/revenue collection systems impacts business competitiveness, frequently in a decisive manner. It is truly the "blood stream" of postal operators and other mail carriers.

While a postal communication system is sometimes considered less efficient than other modern communication systems such as e-mail or fax, it remains the only universal system of message delivery. Moreover, the postal system offers broad bandwidth at a very reasonable cost and all security and legal advantages of paper-based communication that still forms the backbone of the industrial world commercial system.

The development of digital technology resulted in dramatic changes in the methods of mail generation, processing and even delivery. Although we have no specific data, it is probably reasonable to estimate that 80% of the letter mail in the industrial world is generated, finished and processed by computer-driven systems. Almost half of the computers responsible for mail generation and processing are connected to computer communication networks. The payment/postal revenue collection process for mail items generated by such computers is an example of network computing with one unique feature, namely a required (due to pre-payment) physical evidence of payment imprinted or otherwise attached to mail pieces. This evidence of payment is also necessary because of anonymous, widespread and simple access to the postal distribution system that mailers enjoy. A special feature of postal revenue collection is the requirement that all mail pieces are prepaid before mail processing begins. This turns out to be very important from an economic efficiency viewpoint. Detailed economic analysis of possible postal revenue collection alternatives indicates that any reasonable system with the mailer's access similar to existing ones, with comparable security and without prepayment would be hopelessly inefficient [15].

A simplified diagram of a postal communication and revenue collection system is presented in Figure 1. Mailers' terminals connected through the public

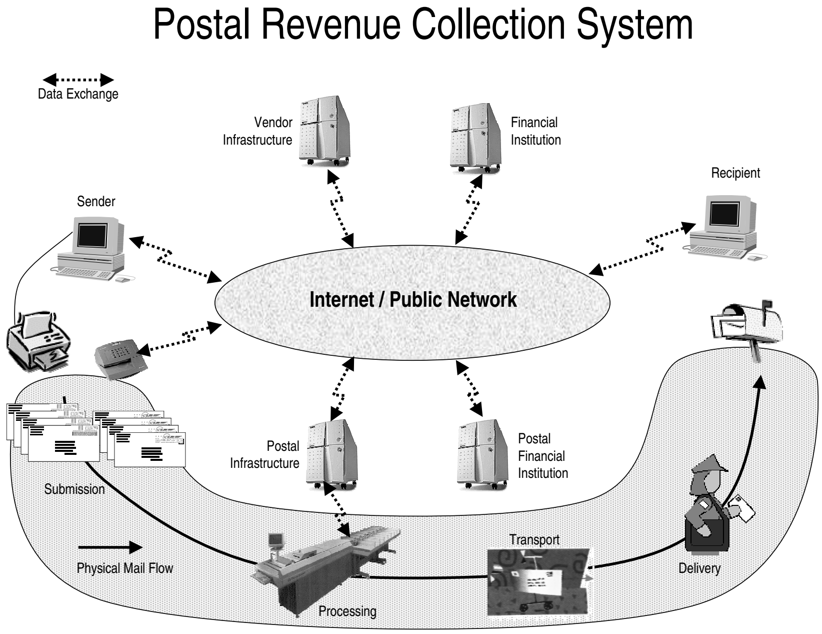

Fig. 1. Postal revenue collection system.

communication network to vendor's (such as Pitney Bowes) and postal infrastructures can be viewed as computational devices equipped with printers. In some cases these devices and printers form one secure tamper resistant unit. This is the case for example of a traditional postage meter. The printer part of the system in this case is dedicated to printing only evidence of postage, which we will refer to below as the Digital Postage Mark (DPM). In some other cases the printer can be a general-purpose office or other (e.g. high-speed) printer that is used for multiple purposes. Another architectural distinction between different possible terminals is whether a mailer's computing device does or does not include a special cryptographic module designed to perform all cryptographic computations required, in particular computations of a cryptographic validation code that must be included in the Digital Postage Mark. In some instances cryptographic computations are performed within a secure remotely located Data Center that forms a part of a vendor's infrastructure. In all cases the system makes use of a special tamper resistant/tamper evident cryptographic module. We refer to this module below as the Postal Security Device or the PSD. The PSD also executes all protected accounting functions which we will discuss below and thus the PSD also serves as the accounting unit. When the remote Data Center is used, the DPM data is transmitted to the mailer's terminal electronically. In both cases the main purpose of the mailer's terminal is to produce and/or finish physical mail items that must carry evidence of (pre)payment in the form of Digital Postage Marks (see Figure 2). The amount of data involved

Fig. 2. Letter with digital postage mark.

and the need for extreme reliability in capturing the DPM's data necessitates the use of two-dimensional bar codes such as Data Matrix or PDF417 code (shown) for physical data representation [2, 1].

Mailers deposit mail pieces into the postal distribution system where the evidence of payment (DPM) is examined and mail is sorted and delivered. The DPM examination involves its capture, digitization and performing of various consistency tests. The main objectives of the DPM examination are fraud detection and assistance in collection of legally admissible evidence that may be required for fraud prosecution.

There are several fairly complex interactions between different parts of the sys tem shown in Figure 1; for example, interactions between the mailer's terminal and vendor's infrastructure. These interactions include enabling and disabling services, postage refills, reporting, billing and audit activities. They are normally organized by execution of special secure communication protocols within a client-server architecture. Although very specialized and challenging, these protocols fall into a class of telecommunication protocols that must meet fairly traditional security requirements. On the other hand, the generation and validation of Digital Postage Marks are clearly unique to postal applications. For this reason this paper is primarily dedicated to cryptographic aspects of Digital Postage Marks.

From a mailer's perspective, the purpose of mailing is the reliable delivery of messages and goods to a recipient; postage prepayment is merely a necessary prerequisite to such delivery. Mailers require postage payment seamlessly and cost-effectively integrated with mail generation and finishing. These processes depend on characteristics of the mailing application, e.g., transaction, correspondence, advertising, and mail volume. For some mailing applications generating and validating evidence of postage is simpler than for others. Several large postal administrations including the United States Postal Service (USPS) have become interested in defining PC-based desktop systems capable of secure generation of the Digital Postage Marks [18, 16]. There are millions of desktop systems that can be used for this purpose and typically mail that is generated by such systems is deposited into street letter boxes. These two circumstances create an important and special application of the Digital Postage Mark. For example in this case the key management advantage of public-key schemes is more pronounced than in other DPM applications due to a very large number of potential users involved. Also, mail collected from street letter boxes is processed differently than other types of mail. For these reasons we restrict our analysis here only to PC-based desktop applications.

The remainder of this paper is organized as follows. Section 2 discusses the basic ideas behind postage payment applications and Section 3 addresses the requirements necessary for a secure cost-effective solution. The optimal mail certificate (OMC) is a vital component in our solution and this is considered in Section 4. A detailed description of its creation and benefits will be given. Fundamental to the verification of a DPM is the cryptographic validation code (CVC) which is effected through a digital signature based on elliptic curve technology. The digital signature mechanism proposed in this paper is new, and is based on the ElGamal signature mechanism having partial message recovery. It is described and evaluated in Section 5. In Section 6 we discuss the merits of the new scheme; in particular, it evaluates how well the scheme meets the requirements outlined in Section 3. The reader is referred to [11] for detailed discussion on the basic cryptographic concepts used in this paper.

2 Fundamentals

The basic idea of postage payment evidencing is quite simple. A verifier can examine a mail piece in order to find evidence that payment (or, more precisely, accounting) for this mail piece has been made. What can constitute such evidence? If the verifier knows that the postage mark imprinted on the mail piece has been produced by a known printing mechanism and that the printing mechanism was securely coupled to an accounting unit in such a way that printing could not have been accomplished without accounting, then, by examining (forensic) properties of the imprint, the verifier can conclude whether accounting (payment) has or has not been made. This is the original principle of postage meters with mechanical coupling of printing and accounting functions. Security of postage meters based on this principle rests on the assumption that forensic qualities of an imprint are sufficient to identify the printing mechanism and legitimate printing mechanisms are tightly controlled. This assumption is obviously incorrect in the case of digital printing produced by PC-based desktop systems. Thus, printing must be controlled in a different way. For example, if the informational content of the DPM is directly indicative of the accounting action, then by examining this informational content the verifier can be convinced that the accounting indeed took place before printing. This can be achieved by having the accounting unit (PSD) exercise full control of a cryptographic private key which must be used to authenticate informational content of legitimate DPMs. This is the fundamental principle of digital postage meters.

Our postage evidencing model includes a sender (mailer) and a verifier (Post). The mailer generates and sends a DPM imprinted on the mail piece to the Post. The Post accepts or rejects the mail piece depending on the consistency of the information in the DPM. An adversary may intercept the DPM and replay it or generate and submit his own DPM based on an intercepted message or independent of it. A mailer may produce some legitimate mail pieces, and become an adversary with respect to other mail pieces. As usual in protocol security analysis, we assume that messages and algorithms are in the public domain.

The four information security objectives critical for postage evidencing are:

- *Data Origin Authentication.* The Post can read the DPM including the identity of the postage accounting device (PSD) responsible for the DPM.
- *Data Integrity.* The Post can detect any alteration of the DPM.
- *Evidence of Fraud.* The Post can produce evidence of fraud, such as a mail piece with a counterfeited DPM or mail pieces with identical DPMs.
- *Confidentiality.* In some cases it may be desirable to protect the confidentiality of certain data elements within the DPM, for example some information indicative of the mailer's mailing activities or the mailer's e-mail or fax addresses that may be required for special services such as confirmation of delivery.

The DPM always contains some plaintext and some form of a digital signature. We call the plaintext the Postal Data (or the PD) and the digital signature the Cryptographic Validation Code (or the CVC). The purpose of the CVC is to satisfy the aforementioned security objectives. The CVC can be computed using symmetric or asymmetric cryptographic transformations. If a public-key mechanism is used then confidentiality is more difficult to maintain.

All data in the plaintext is signed but the question is: What data elements should be included in the plaintext (PD)? In order to achieve origin authentication, we need a unique postage accounting device identification number and a message identification number; for example, a serial mail piece count or the value of the ascending register in the accounting unit. (The ascending register in the meter keeps track of the value of the postage processed to date.) The integrity of the postage value must always be protected by inclusion of the postage amount. These three elements represent a minimal set. This minimal set has approximate size of 10 bytes. Depending on the verification strategy, additional elements may be included into the PD; for example, date and sufficiently precise delivery address information. The date and delivery address information can be represented using another 10 bytes. Thus, the total size of the PD is on the order of 20 bytes. Of course, additional elements can be included depending on the application. For example, the recent USPS document Information-Based Indicia Program recommends the size of the PD equal to 49 bytes [18].

If for some reason the delivery address is not available for inclusion in the DPM then the DPM cannot contain any data specific to a given mail piece. Then a genuine, legitimately pre-paid DPM can be duplicated and imprinted or otherwise attached to another mail piece. In this case, duplicates must be detected and intercepted by the verification system. For example, an attacker may send multiple duplicates to one office building housing many different recipients. Therefore, the system must detect duplicates (this is actually true even with the delivery address information in the PD, although in this case the economic attractiveness of the attack is greatly diminished). Duplicate detection is considerably more effective, especially for a less than 100% sampling strategy, if the Post restricts the valid mail deposit to a specific date and specific geographic area. This suggests that the PD should also include the date and the postal code or the name of the allowed geographical deposit area. What is important for us here is the estimate of the total amount of data in the PD that requires protection. Our analysis as well as several draft specifications [16]issued by postal authorities around the world indicates the size of the PD portion of the DPM is between 20 and 50 bytes. We shall use this estimate later for comparison of different possible options for computing the CVC.

Our main interest is the optimal design of the DPM, particularly the CVC portion of the DPM. In the next section we outline and explain basic optimality criteria critical for our analysis.

3 Requirements

The DPM design is subject to a set of intuitively desirable requirements given below.

1. *Total break resistance.* The CVC must possess key compromise related cryptanalytic strength above a certain (commonly accepted) threshold, for example 2^{80} operations.
2. *Selective forgery resistance.* The CVC must possess selective forgery related cryptanalytic strength above a certain application dependent threshold, for example 2^{40} operations. This usually means that the best known algorithm to forge a signature (the CVC) using publicly available information requires at least 2^{40} operations and has to be repeated for each new CVC to be forged.

The threshold number (e.g. 40) is a function of the monetary amount to be gained by the forger and the amount of computational resources available to her. In the case of the DPM the monetary amount is usually very small (e.g. $0.33 in USA). Assuming that a powerful PC is the only computational resource available to the forger then this is a reasonable estimate for the security level. Minimal running time of several hours on such a PC for forging a single CVC may deliver the desired deterrence effect and may be acceptable from a security view point.

3. *Minimal size.* It is critical to keep the size of the CVC to a minimum due to severe limitations in the space on a mail piece available for the DPM's physical representation and the need to capture and interpret the DPM in a highly reliable fashion within a relatively short period of time. This requirement is unique for postal applications and it has the most profound implications for the viability of the entire revenue collection approach. One must keep in mind here also that optical readers for DPMs are more accurate when the physical size of the DPM is small and that the DPM is aesthetically more appealing if this is the case.
4. *Signature size inflation resistance.* The CVC size inflation due to improvements in cryptanalytic algorithms and computing power also should be kept at a minimum. This means that the cryptographic algorithm used to create the CVC (digital signature) should be such that the size of key (and consequently the size of the signature) should increase at a minimum pace (as cryptanalytic algorithms improves) to maintain a required level of cryptanalytic strength.
5. *Computational efficiency.* Computational performance of the CVC generation and verification processes should be appropriate to match performance of the fastest mail generation and processing (verification) equipment. In practice this means a speed of up to ten CVC generations per second for autonomous systems, a few hundred CVC generations for client-server based systems and up to 20 verifications per second will be necessary.
6. *Self-sufficiency.* It is very desirable to make the DPM contain all the information required for verification. This means that it is desirable to have the verifier perform the verification process without a need for continuous access to external data sources. In other words, the verifier should be able to determine internal consistency of the information in the DPM based solely on the DPM data. In the scenario under consideration this means that the authenticity of the mailer's public key can be determined from the DPM itself.
7. *Multiple test.* It is desirable to have the verifier perform additional tests based on the DPM information, which can further reduce risk of misusing postal funds. These additional tests may include verification of certain parameters contained in the PD against pre-defined criteria stored at the verifier or present in the DPM. For example, certain devices may have the privilege to create and print postage evidencing information associated with an expiration date or may have restricted privilege to print postage value above a certain threshold or other similar restrictions.

8. *Confidentiality.* It is desirable to protect confidentiality of certain data elements within the DPM. This means that these data elements should not be present in the DPM as a part of the plaintext, but should be recoverable from the CVC only by an authorized party such as the Post itself or its designated verification/data processing agents.
9. *Economic efficiency.* The cost of the entire DPM generation-verification system should be minimal to enable broad access and efficiency to the mailers and the Posts.

Finding solutions that satisfy all the requirements in this list is difficult. For example, the first, second and sixth requirements above appear to be in direct contradiction with the third requirement, if one is forced to use a public-key scheme based on a standard certificate system such as X.509 [9].

The sixth requirement outlines a very important consideration. For postal revenue collection applications this requirement makes key management systems highly effective and that, at least in the opinion of one of the authors, provides the only true justification for using public-key cryptography.

The first requirement is the ubiquitous security requirement which must be satisfied for any system to be sound. In the context of the elliptic curve schemes (to be discussed in the next section) that rely on the difficulty of the discrete log problem, this translates into the requirement that the size of the group of points on elliptic curve should be at least 2^{160}. This is motivated by the value of the work factor for the known best algorithm for computing discrete logarithms on elliptic curves and it is approximately equivalent to the work factor required to break a 1024-bit RSA scheme by the best algorithm known for factoring large composite numbers.

Table 1 lists key size estimates. The estimates for RSA security were based on the security estimates provided by NIST for the revised Digital Signature Algorithm [3] and using the fact that the best algorithms known for integer factorization and the (ordinary) discrete logarithm problems require approximately the same amount of resources. The estimates for ECC security were provided by NIST [12].

Table 1. Comparing ECC and RSA key lengths for same levels of security.

Symmetric cipher key length	Example algorithm	ECC key length for equivalent security	Rough estimate of RSA key length for equivalent security
80	SKIPJACK	160	1024
112	Triple-DES	224	2048
128	128-bit AES	256	3072
192	192-bit AES	384	7680
256	256-bit AES	512	15360

In the next sections we present a digital signature scheme with partial message recovery and what we believe to be the optimal certification mechanism that satisfies the sixth requirement. It is our opinion that this scheme delivers the best balance between the contradictory requirements given above and thus represents the optimal choice among all known systems. For the convenience of the reader we summarize the requirements detailed in Table 2.

Table 2. Requirements.

Requirements	Brief description
1. Total break resistance	Resistance against compromise of secret keying material.
2. Selective forgery resistance	Resistance against forging signatures without knowledge of the secret key.
3. Minimal size	DPM should be as small as possible for both physical and aesthetic reasons.
4. Signature size inflation resistance	Key sizes should not expand dramatically as computing and algorithmic power increases.
5. Computational efficiency	Generation and verification of CVC should be as efficient as possible.
6. Self-sufficiency	DPM contains all information necessary to verify CVC.
7. Multiple test	Use of additional information besides CVC to validate DPM information.
8. Confidentiality	Ability to provide confidentiality on some data elements in DPM.
9. Economic efficiency	Minimize overall cost of DPM generation and verification.

4 Optimal Mail Certificates

In this section and the next we describe a simple and elegant scheme that goes a long way to satisfy our sixth requirement. A brief explanation is in order. When the Post verifies the CVC it can retrieve all vital information (e.g. certificate, public key and signature) from the DPM itself. Proposals have been put forward where the public key and certificate of the PSD (mailer's terminal) are stored in a database and retrieved at verification through an identifier contained in the DPM. Such proposals have the disadvantage that a large database is required but proponents argue that the bandwidth saved in the DPM is worthwhile. It is our contention that the new scheme we propose has the same bandwidth requirements but removes the necessity of the large database.

The setup for the scheme is as follows. Let P be a public point of order n in the group of points of the elliptic curve $E(\mathbb{F}_q)$ over the finite field $\mathbb{F}_q$ (the total number N of points on the curve is divisible by n). Minimal size for n is approximately 20 bytes. (The reader is referred to the book by Menezes [10] for definitions and terminology used for elliptic curve cryptosystems.)

We assume that the Post either functions as a Certificate Authority (CA) or uses one of the established Certificate Authorities. In its capacity as a CA the Post generates a random integer c between 0 and n. The integer c is the postal system wide private key. The corresponding postal system wide public key is $B = cP$. The secrecy (confidentiality) of c against cryptanalysis is as usual protected by the difficulty of elliptic curve discrete logarithm problem.

Each mailer's terminal A has an identity I_A. The identity I_A may contain a number of additional parameters and attributes besides strictly identification information for the mailer's terminal, its PSD and mailer's identity itself. These parameters depend on application requirements and may include the expiration date, allowed maximum postage value or allowed maximum number of DPMs to be produced by the terminal, an indication of allowed geographical area where mail items produced by the terminal can be deposited etc. The identity I_A is assigned prior to the beginning of operations by the Post or a registration authority such as a vendor trusted by the Post. The identity I_A provides a natural mechanism to satisfy our seventh requirement. It is printed in the PD portion of DPM in plaintext.

The mailer's terminal A generates a random positive integer $k_A < n$, then it computes the value $k_A P$ and sends this value to the Post. It should be noted that this phase could in fact be done using a long term private/public key pair from a more traditional X.509 certificate key pair. This can be done once for a given period of time or for a given number of authorized DPMs that can be generated by the terminal.

The Post generates a random positive integer c_A smaller than n and then computes the point γ_A on the curve

$$\gamma_A = k_A P + c_A P.$$

We call the value γ_A an "Optimal Mail Certificate or OMC" in mailing applications, but γ_A is more commonly referred to as an implicit certificate [8, 17].

Next the Post computes a value

$$f = H(\gamma_A || I_A),$$

where H is a hash function such as the SHA-2 and "||" as usual denotes the operation of concatenation. At this point various restrictions on the data included in I_A and in the DPM can be tested. The Post then computes its input m_A to the mailer's private key a as follows:

$$m_A = cf + c_A \bmod n$$

and sends values γ_A, m_A and I_A to the mailer's terminal A. This portion of the protocol is executed once for a period of time prior to mail generation/verification operations.

The mailer's terminal A computes its private key a and its public key Q_A as follows:

$$\begin{aligned} a &= m_A + k_A \bmod n = cf + k_A + c_A \bmod n \\ Q_A &= aP = cfP + \gamma_A = fB + \gamma_A. \end{aligned}$$

This is also done once for a period of time determined by security and application considerations.

The private key a is used by the terminal A to compute the validation code CVC from the plaintext PD using a digital signature with partial message recovery based on the Nyberg-Rueppel [13] scheme which is a variant of the well studied ElGamal signature mechanisms. This will be described in the following section. Observe that the private key a is a function of a postal system wide private key c and mailer-specific postal private parameter c_A as well as the mailer's private parameter k_A . Note also that the CVC verification key Q_A is a function of only the public parameters and is computable from the OMC γ_A, postal system wide public key B and the value f of the hash function.

The DPM verification process can be organized as follows. After capturing the DPM data it is parsed into the PD and the CVC portions. The OMC γ_A and identity I_A are used to compute the hash value f. Then the verification key Q_A is computed using the postal public system wide key B and the OMC γ_A. Then the CVC is verified using a version of EC ElGamal signature verification described in the following section with the verification key Q_A serving as the public key. It has been shown [6] that under the random oracle model this procedure is as secure as the elliptic discrete logarithm problem.

The OMC γ_A is simply a single point on the elliptic curve $E(\mathbb{F}_q)$ which has the size of the underlying field element plus one bit (if a point compression technique is used), which is in our case approximately 20 bytes. When included in the DPM the OMC greatly simplifies key management at the expense of about 20 bytes of overhead. Excluding plaintext, the size of the standard ECDSA signature scheme [4] with the OMC included is only 60 bytes compared to 128 bytes of RSA signature alone. We discuss size implications of different schemes in more detail in the concluding section of the paper.

5 Cryptographic Validation Code as a Digital Signature with Partial Message Recovery

In this section we describe a new digital signature scheme with partial message recovery designed to satisfy our third and eighth requirements. Combined with the optimal mail certificate scheme of the previous section this system delivers the known best overall solution. Appropriate comparisons and discussion are given in the concluding section.

The partial message recovery scheme to be described below is similar to one proposed in the draft standard ISO/IEC 9796 Part 4 but is computationally and bandwidth more efficient.

In the DPM application all messages to be signed have a fixed short size typically smaller than 160 bits. Under this assumption we will show that a signature scheme with partial message recovery seems most appropriate.

We first divide the plaintext PD into two parts, namely a part C which represents data elements that require confidentiality protection and that can be recovered during the verification process from the signature and a part V

that contains data elements presented in the plaintext within the DPM. This means PD $= C||V$. The integrity of the data elements in V is still protected since V is also signed. This separation of the PD into parts fits our application almost perfectly. Due to a variety of traditional, marketing, postal accounting, appearance and human readability requirements some data elements in the DPM must be present for immediate examination (e.g. by the recipient). These data elements include date, postage value and the postal code of location where the mail piece originated. These elements are candidates for the part V. Other data elements such as the value of a serial piece count, the value the ascending register, e-mail address, telephone or fax number of the sender and the like can naturally form the part C. These data elements allow for a cost effective organization of a number of special postal services such as a proof of deposit and delivery and mail tracing. This helps to satisfy our ninth requirement.

The signature generation algorithm for the message PD $= C||V$ begins as usual with the generation of a random positive integer $k < n$ by the mailer's terminal A. The terminal performs the following computations:

1. $R = kP$; R is a point on the curve that is formatted as a bit string for the transformation defined in step 2.
2. $e = Tr_R(C)$, where Tr_R is a bijective transformation parametrized by R and designed to destroy any (algebraic) structure that C might have. Transformation Tr may be a symmetric-key encryption algorithm such as DES or simply the exclusive-or (XOR) operation if C is at most the length of R. Secrecy of R is protected by the difficulty of the discrete log problem and a random choice of k.
3. $d = H(e||I_A||V)$, where H is a hash function and I_A is the identity of the mailer's terminal.
4. $s = ad + k \bmod n$, where a is the private key of the terminal A computed as described in the previous section.
5. Pair (s, e) is the signature (the validation code CVC) and it is presented for verification in the DPM together with the portion V of the plain text PD.

Note that step 2 is computationally efficient if the size of C is less than or equal to the size of R and the transformation Tr is exclusive-or. For many applications of DPM it is true that the size of C is less than or equal to 20 bytes (see Section 2 with the estimates for the size of PD).

The DPM verification process begins with the capture of the DPM from a mail piece and parsing the DPM data into the values I_A , CVC=(s, e), V and γ_A. Then a postal verifier performs the following computations:

1. $Q_A = fB + \gamma_A$, where Q_A is the mailer's terminal public key as described in the previous section and B is the system-wide postal public key; note that B does not need to be known outside of the postal verification system.
2. $d = H(e||I_A||V)$.
3. $U = sP - dQ_A$
4. $X = Tr_U^{-1}(d)$, recovering a new value X by the inverse transformation Tr^{-1} parametrized by the value U.

5. Check redundancy of X and if X has required redundancy (e.g. 40 bits) declare $C = X$ and accept the signature as valid.

In postal applications the confidential data C is normally quite redundant. This means that components of C must have specific meaning known in advance by the verifier. For example, the value of the e-mail address must be of a specific form or the ascending register must be larger than a certain value etc. Of course, additional redundancy can be added as desired, but not without a price to be paid. The size of C and efficiency of the computation in step 2 of signature generation can be adversely affected. Trade-offs between the amount of effort to forge a signature and the size of C must be carefully evaluated to provide for overall optimal economic efficiency.

Confidentiality of C is protected only if the postal verification public key B cannot be easily obtained outside of the postal verification system. This is fortunately the case since there is no good reason to maintain access to B for anything other than verification applications. It is interesting to point out that in this scenario a public-key scheme is being used as if it were a private-key (or symmetric-key) scheme. The advantage gained, of course, is that even if confidentiality is lost, integrity is maintained.

If the plaintext PD is small, then the PD can be "hidden" within the signature in its entirety. Importantly, our scheme allows for particular efficiency if there is no need to present the "open" portion V of the PD in the DPM twice. Due to a very high DPM readability requirement (typically 99.5%), the open portion V may need to be represented both in a human-readable and machine-readable formats (bar code). If the human readable format allows for a high readability, for example by employing a specially designed OCR font of appropriate dimensions and with appropriate formatting, then the size of the DPM can be further reduced.

For a detailed analysis of the security of this signature scheme, see [6].

6 Discussion and Conclusion

The fundamental information security based approach to DPM was developed in the early eighties within Pitney Bowes by Clark, et al [7] . In 1987-1989 J. Pastor, also from Pitney Bowes, developed and adapted for mailing applications several critical aspects of digital signatures, including a signature based on elliptic curve techniques [14]. In 1996-1999 the USPS published a series of draft DPM specifications based on public key schemes. None of these efforts however achieved optimization of the DPM design. We believe that the signature scheme described in the Section 5 when used together with the optimal mail certificates delivers the optimal choice in view of the requirements formulated in Section 3. The first two requirements are necessary pre-requisites for security of a revenue collection system. The second requirement brings security into the economic context of the entire system. It takes into account not only difficulties that cheaters must face, but also, and equally important, it attempts to factor in the economic attractiveness of the contemplated fraud. The third requirement is critical for the viability of any system designed around physical representation of data required for verification. Severe limitations of space available for the DPM dramatically restricts usefulness of the otherwise very effective approach (imagine for example small post cards, "thank you" notes and the like). Table 3 demonstrates savings in the DPM size afforded by our solution in comparison with other possible designs (we assume as usual that all signature schemes included in the table are approximately equivalent in their resistance to a total break of a 1024-bit RSA signature and that the certificate signature and the data signature schemes are identical). Note that RSA as well as ElGamal signature schemes can also be used in message recovery mode. This would reduce the size of the DPM compared to the case of RSA and DSA schemes with appendix given in the table. We have chosen to present the table in this form because some postal administrations, for example the USPS, recommend the use of standard RSA and DSA signatures with appendix only [18]. The numbers in the table were computed as follows:

1. For RSA we assume a 1024-bit modulus and a signature scheme with appendix (as specified by the USPS [18]. The certificate is assumed to contain only the public key and the CA signature.
2. For the DSA the modulus is taken to be 1024 bits, the signature size is as specified by the DSA itself. The certificate is assumed to contain only the public key (128 bytes) and the CA signature (40 bytes).
3. For the ECDSA the order of the elliptic curve is approximately 20 bytes, the signature is 40 bytes (similar to the DSA) and the certificate contains a 20 byte public key (assuming point compression) and a 40 byte CA signature.
4. For the EC with MR we assume the elliptic curve order is approximately 20 bytes, the signature is 20 bytes (assuming no additional redundancy is added to the message) and the certificate consists of a 20 byte public key and a 40 byte CA signature.
5. For the EC with MR and OMC, we assume a 20 byte elliptic curve, a 20 byte signature (assuming no additional redundancy for the message) and a 20 byte OMC.

Note: In case of EC with MR and EC with MR and OMC if the message contains no inherent redundancy (or little) one may have to add up to 10 bytes of additional redundancy. In other words, the totals given in the last two columns might have to be increased by up to 10 bytes. As discussed earlier, messages in this environment typically contain sufficient redundancy for the intended application. It should however be mentioned that one must consider the additional cost required during the verification process to check message redundancy and do appropriate trade-offs with time and space. One possible option is to put part of the OMC in the PD (if there is room) and make this part of the V portion of the PD.

Table 3 demonstrates that the DPM size can be reduced quite dramatically. Potentially even more important from a long-term view point is the fourth requirement. The key/signature size for some digital signature schemes is expected to increase by 20-30% in the next 5 years due to improvements in algorithms and computing power. The relative strength per bit of the key/signature is a serious

Table 3. DPM size (in bytes) using different protocols.

	RSA	DSA	ECDSA	EC with MR	EC with MR and OMC
PD	20	20	20	(20)	(20)
CVC	128	40	40	20	20
Certificate (min. size)	256	168	60	60	20
Total DPM	404	228	120	100	60

consideration. In this context, elliptic curve techniques that we have adapted for the DPM application here so far have proven to be more robust than others.

As mentioned earlier, proposals have been put forward to remove the certificate and public key from the DPM and store these in a central database. For RSA and DSA this would leave a DPM whose size is 148 and 60 bytes respectively. Comparing this with EC with message recovery and OMC one has the same size DPM as with the DSA and still requirement 6 is met.

The sixth requirement aims at greatly simplifying key management for the DPM verification process [5]. The need for the Post to coordinate public keys for millions of users having their systems supplied by multiple independent providers represents a significant burden on the system. The seventh requirement, although not directly related to the choice of cryptographic mechanism for the DPM, can be satisfied in a particularly simple way through the use of optimal mail certificates. The eighth requirement can be met by any signature scheme with message recovery provided the OMC is used.

Finally a few words about economic effectiveness of the DPM generation/ verification system which constitutes the last requirement. This is the most important and in fact permeates all other requirements. The non-digital revenue collection system employed in many countries today is quite functional. Moreover, it is probably true that a revenue collection system based on an annual estimated tax can be functional as well. The system based on the DPM must be more effective than other alternatives, otherwise it can not survive. So our first eight requirements are in fact all efficiency requirements aiming at either reducing losses due to potential fraud or reducing cost of DPM generation and verification. For example, minimal size DPM are not only critical because of limitations in physical space, but also contributes to better readability and less cost of consumables for the printing process as well as allows to better provide for many value added services. Similarly, computational efficiency allows reduction in the cost of components required for the DPM generation and verification.

References

1. AIM USA-1994, *Uniform Symbology Specification PDF417.*
2. ANSI/AIM BC11-1997, *International Symbology Specification - Data Matrix.*
3. ANSI X9.30, *Public Key Cryptography for the Financial Services Industry: Part I: The Digital Signature Algorithm (DSA) (Revised)*, draft, July 1999.
4. ANSI X9.62, *Public-key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)*, 1999.

5. T. Biasi, R. Cordery, S. Joshi and L. Pintsov, "Digital postage mark verification", *Proceedings of International Conference on Mail Technology-Tomorrow's World*, Brighton, UK, 1999, pp. 199-211, published by Professional Engineering Publishing Ltd, for the Institution of Mechanical Engineers, Bury St Edmunds and London, UK 1999.
6. D. Brown and D. Johnson, "Formal security proofs for a signature scheme with partial message recovery", Technical report CORR 2000-39, Dept. of C&O, University of Waterloo, 2000. Available from http://www.cacr.math.uwaterloo.ca
7. J. Clark, A. Eckert, D. Warren, *System for the printing and reading of encrypted messages*, U.S. Patent 4,641,346, February 1987.
8. M. Girault, "Self-certified public keys", *Advances in Cryptology: Eurocrypt '91*, 1991, pp. 490-497.
9. ITU-T REC X.509 (revised), *The Directory-Authentication Framework*, International Telecommunication Union, Geneva, Switzerland, 1993 (equivalent to ISO/IEC 9594-8:1995).
10. A. Menezes, *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers, Boston, 1993.
11. A. Menezes, P. van Oorschot and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.
12. National Institute of Standards and Technology, *Recommended Elliptic Curves for Federal Government Use*, May 1999; revised July 1999. Available at http://csrc.nist.gov/encryption.
13. K. Nyberg and R. Rueppel, "A new signature scheme based on the DSA giving message recovery", *1st ACM Conference on Computer and Communications Security*, ACM Press, 1993, pp. 58-61.
14. J. Pastor, "CRYPTOPOST$^{\mathrm{TM}}$: A cryptographic application to mail processing", *Journal of Cryptology*, **3** (1990), pp. 137-146.
15. L. Pintsov, S. Joshi and T. Biasi, "Transaction Cost Economics of Postage Payment and Mailer-Post Interface", in *Emerging Competition in Postal and Delivery Services* (Editors M. Crew and P. Kleindorfer), pp. 295-307, Kluwer Academic Publishers, 1999.
16. *Postage Indicia Standard for Canada Post*, Version 1.2, draft, April 2, 1999.
17. M. Qu and S. Vanstone, "Some new efficient implicit certificate schemes", Certicom Research, preprint.
18. USPS Information Based Indicia Program (IBIP), *Performance Criteria for Information Based Indicia and Security Architecture for IBI Postage Metering Systems (PCIBISAIIPMS)*, draft, August 19, 1998. http://www.USPS.com/IBIP