# A RISK MINIMISATION FRAMEWORK FOR ELECTRONIC COMMERCE

*Denis Trèek*
*"Jožef Stefan" Institute*
*Jamova 39, Ljubljana,SLOVENIA*
*Tel.: +386 61 177 3379*
*Fax: +386 61 219 385*
*e-mail: denis.trcek@ijs.si*

*Abstract. Various kinds of digital commerce depend on digital signatures, which are based on public-key cryptography and one-way hash functions. However, one-way hash functions have properties, which can be exploited to subvert security service. A framework in this paper gives a way to minimise such risks with paying attention to proper structuring (and consequently coding) of electronic documents and the context of their usage.*
*Keywords: security services, public-key cryptography, digital signatures, one-way hash functions, electronic commerce.*

## 1. Introduction

**The era of global digital communications is emerging rapidly. It is no more a question if cyber-space will serve user community for sensitive business tasks (such as financial transactions) - it is more a matter of such escalation on a global scale. The possibilities for electronic business on Internet range from "ordinary" EDI standards [EDI93, X.12] to newly emerged proposals like micro-payments systems [Bell95] and more general and complex systems like, e.g. SET [Mast96], which were triggered with the explosion of WWW. Each of these systems is aimed at specific purposes and is architecturally designed accordingly. For example, micro-payments systems (where the value of a transaction is in a range of transaction processing costs) are based on keyed Message Authentication Codes (MACs [Tsu92, Gal91]). SET, on the other hand, tries to provide a general framework for secure transactions over the Internet. And finally, classical EDI is designed to provide means for properly formatted data that would enable automatic processing of business documents.**

**As every business transaction involves exchange of a business document, a large diversity of electronic documents (e-docs) will be used to accommodate a variety of such needs. Of course, most of these documents will have to be digitally signed. This is usually achieved by taking a hashed value of a message and encrypting it with a secret key to prove authentication and integrity of a particular message.**

**However, ideal hash functions and their practicable implementations have properties that can be exploited to falsify a digital signature. Most efforts went into refining and new design of these functions to prevent such weaknesses. The approach in this paper is opposite. Properties of cryptographic mechanisms are taken as they are, while risk minimisation is tried to be accomplished with a proper design of electronic documents and their appropriate usage in a certain communication context. It will be shown that the basic principles behind these attacks are strongly related to the data being transformed (which, for example, is exactly the task of EDI). Analysis of these principles serves as a starting point for guidelines, which make e-doc transactions more secure. Therefore the proposed solutions are not in the domain of cryptography but in the domain of e-doc design and their appropriate usage with respect to attacks on one-way hash functions. Solutions are proposed on the application level that directly manipulates e-docs prior their delivery to the communication stack.**

**The paper is organised as follows. In the second section there is a description of the problem. Next, some background of attacks on one-way hash functions is given with relation to e-docs. This is a starting point for a risk minimisation framework. There is a conclusion in the last section, while a short appendix gives a basis for common understanding of terms, used in this paper.**

## 2. Some background of attacks on one-way hash functions

**Recent advances in cryptography showed ways to attack three widely used one-way hash functions - MD2 [Kal92], MD4 [Riv90] and MD5 [Riv92]. Besides plain message digest calculation with MD2 [Rog95], MD4 [Dob96a] and MD5 [Dob96b], this also includes e.g., MD4 usage for MACs [Pre95].**

**It turns out that it is beneficial to treat application of cryptographic techniques and e-docs as a whole and this is the main motivation behind the paper. The proposed framework takes into account common principles behind attacks on MD2 and M5, which are almost exclusively based on finding collisions [Rob96].**

### *2.1. Idealised one-way hash functions*

**An ideal strong one-way hash function has the following properties:**

1. **Pre-image resistance: for a given** $y$ **it is computationally infeasible to find such** $x$ **that** $f(x) = y$**.**
2. **Second pre-image resistance: for a given** $x_1$ **it is computationally infeasible to find** $x_2$ **with the same hashed value, i.e.** $f(x_1) = f(x_2)$**.**
3. **Collision resistance: for any** $x_1$ **it is computationally infeasible to find** $x_2$ **with the same hashed value, i.e.** $f(x_1) = f(x_2)$**.**

**One-way hash functions map a universal set of texts to a set with a finite cardinality, thus collisions will always exist. More precisely, as long as all elements from a non-finite universal set will be mapped to elements of a finite one, collisions can not be avoided. The problem is to ensure computational infeasibility for such events. Unfortunately, even the existence of collision-resistant hash functions (in an idealised strict sense) is still an open question [Dob96b].**

**For the time being, probability based attacks could turn out to be feasible. Basically, two conceptually different approaches are possible***: fixed-date paradox collision* **and** *birthday-paradox collision.* **In the first case a pre-defined text is to be replaced by another one, where both of them hash to the same value. In the second case an arbitrary harmless text from set** $A$ **is tried to be matched with a harmful one from set** $B$**, where both of them again hash to the same value. Exact derivation gives surprisingly high probability for the second case and a low probability for the first one (see appendix). Thus a birthday-paradox attack has to be blocked as much as possible. Ideally it should be reduced to a fixed-date collision attack.**

**Fixed date collisions present a bottom line of security for digital signatures produced with one-way hash functions. Consequently, the solution should be as close to this principle as possible. For easier referencing in the rest of the paper, the main points from appendix can be rephrased as follows (**$A$ **is a domain and** $B$ **is a range of a one-way hash function):**

1. **The probability of an input to be hashed to a particular output is equal for each input.**
2. **The cardinality of output range (number of possible hashed values) should be as large as possible with respect to the available processing power.**
3. **The number of acceptable elements from** $A$ **should be as small as possible.**

**(ref.2.1)**

**The first statement is the requirement for the uniform distribution. Together with the second requirement it is related to a pre-image and a second pre-image resistance. Looking at the fixed date collision and the birth-day paradox collision, it can be seen that both kinds of the attacks depend on the cardinality of the range. Additionally, birthday paradox depends on the cardinality of sets *A*.**

### *2.2. Specific implementations of one-way hash functions*

**Cryptographic advances in area of breaking strong one-way hash functions will be considered here. Two of the most recent papers in this field ([Dob96a] and [Dob96b]) will serve as a basis for our framework. Their main achievements can be roughly summarised as follows:**

1. **Searching for a collision of a hash functions' s compress function is a probabilistic process.**
2. **Only a sub-part of a particular message is needed to forge the entire message - the unmodified sub-part is simply appended because of the iterative nature of the MD4 function.**
3. **An attack starts with the design of two e-docs with the two different values (which could be the amount of money to be paid). Next, appropriate random bits are changed in both e-docs to find a collision for these different values.**

**(ref.2.2)**

**Although of a probabilistic nature, the above mentioned attack is based on function' s internal structure. As such it should not be over-simplified by being treated all the way in a generalised manner, as described in the appendix (assumptions are not the same). Nevertheless, practicable implications, summarised in ref.2.2, can be taken into account for the framework.**

## 3. General guidelines for risk minimisation

**To cope easier with the rest of the paper and for the purpose of this framework, the following definitions are introduced.**

1. *e-doc is a sequence of fields, where these fields are divided into fixed ones (with only one possible value) and variable ones (with two or more possible values).*
2. *The field topology of an e-doc is determined by its length (number of fields) and sequence composition (chaining of these fields).*

**It is natural to treat an e-doc as a sequence. Every message, when transmitted over the network, is a sequence of bits. Moreover, when hashing a particular message, this message is also initially treated as a sequence. Thus the problem fits well in the context of the appendix.**

**The second definition is introduced to emphasise the importance of the placement of the particular (semantically interchangeable) fields within e-doc sequence. This will be discussed in more detail in the following subsections. Text fields will be denoted with square brackets. Fixed ones will be written in plain italics, while variable ones will be written in underlined italics.**

**E-doc topology can be very complex. In this framework, only topologies of the same length will be considered, where fields will be treated as the smallest units. Thus analysis on a sub-field level (i.e., appropriate coding) is to be addressed in the future work.**

### *3.1. Fields' values redundancy*

**A particular e-doc family definitely has to consist of some variable fields. However, due to the reasoning behind ref.2.1.3 (and also ref.2.2.3), a reduction of fields' values narrows in general the range of possible values for a birthday attack. Therefore it is wise for these fields to have only few incremental values.**

### *3.2. Message fields' redundancy*

**A particular e-doc family should consist of as few variable fields as possible. It is not only efficient to design e-docs in a highly unredundant form because of a cryptographic overhead, bandwidth requirements, etc. The smaller the message is, the smaller is the cardinalitiy of sub-set in *A*, which can be used for a birth-day attack (ref.2.1.3). Message fields' redundancy is required also because of the ref.2.2.3, e.g. to prevent attacks, where the text would be pumped with the space/back-space combinations. This would "tune" the message for a birth-day attack, as a template presentation part of the software could hide such combinations from a signer.**

### *3.3. Distinguished structuring*

**Distinguished structuring is related to the previous two requirements. It deals with the problem of the topological overlapping between two different families of e-docs. More precisely, although a particular family of e-docs can be structured in line with the above requirements, it may happen that it has identical topology to another e-doc family. Thus fields, that are playing roles of the fixed ones within one family, may turn out to be semantically interchangeable with the fixed fields from another family. Therefore they become variable fields and expand the range for a birth-day attack. To illustrate the problem, assume two families of documents[1], one for the money-transfer orders and one for the money transfer confirmations:**

*[The signer] [orders] [the transfer] [of US$] [$X_1$] [from account #] [$Y_1$] [to account #] [$Z_1$.]*

*[The signer][confirms] [the receipt] [of US$] [$X_2$] [from account #] [$Y_2$] [to account #] [$Z_2$.]*

**The example shows that there should be no topological overlapping between the different families of e-docs. This is hard to ensure for complex documents - EDI, for example, allows structuring with various levels of enveloping using functional group and transaction set headers and trailers. According to ref 2.2.2, if a collision is found for a particular transaction set segment, a number of functional groups can be faked.**

**Thus e-docs should be very carefully designed. A design should start with a simple message, which is evaluated on the basis of each new field that is added.**

### *3.4. Semantics equivalency with different syntax*

**Another useful way to prevent birthday paradox collision is to form (for each transaction) a document, which consists of two semantically equivalent sub-documents with a different syntax. Assume an e-doc, where John M. confirms the withdrawal of $ 100 from his account as a flat monthly fee for various services (e.g. for receiving weekly transaction logs, having access to an automated voice-based bank account responder, etc.):**

*[The signer] [John M.] [agrees with the withdrawal of] [$100] [as a monthly fee for CT Bank services.]*

*[A monthly fee for CT Bank services in amount of] [$100] [is confirmed by signer] [John M.] [ to be withdrawn from his account.]*

**Each of these sub-documents is signed separately. If an attacker finds a collision for one of them, the second one will cause the transaction to be cancelled.**

---

[1] There are many known families of documents in EDI, e.g. Request for Quotation (840) [X12-840], Purchase Order (850) [X12-850], etc.

**Note that such solution could have legal implications. Slight semantic differences in two messages may arise a question of which one should serve as the basis in case of the legal disputes. A way around might be to define one of them as the primary document for legal disputes regarding the semantic of the message. The second one should serve as a technical addition for proving the correctness of digital signature in the first e-doc.**

### *3.5. Communications' context dependency*

**It is worth to emphasise that the resistance of e-docs to attacks should be carefully evaluated in the context of a certain communication. Assume again the above e-doc:**

*[The signer] [John M.] [agrees with the withdrawal of] [$100] [as a monthly fee for CT Bank services.]*

**There are two variable fields in this message. However, for a signer John M. the variable field is actually only the one that specifies the amount (the fourth field in the above example). Any modifications of the second field are in his favour, which is completely different with the bank - the second field matters a lot. Thus for John M. only a fixed-date paradox collision applies, while for the bank a birthday-paradox collision applies. To change the context in which the bank is vulnerable to a birthday paradox based attack, a procedure could be set up, where the signer is asked to sign a confirmation digitally:**

*[The CT Bank asks] [John M.] [to confirm the withdrawal of] [$100] [as a monthly fee for its services.]*

**In this context, variable fields (the second and the fourth one) become fixed and only a fixed date collision is possible for an attacker.**

**Another example of communications' context dependency is the threat described in ref.2.2.3. If someone convinces the other party that random bytes are there for "doing nothing", the range of acceptable messages for him/her is enlarged (as well as the range for unacceptable messages). The principle behind this attack is again a birth-day paradox[2]. It can be concluded that exposition to a certain kind of attack is directly related to a communication context.**

### *3.6. e-doc message segmentation*

**To enforce ref.2.1.3, a complex e-doc structure could be split into small chunks, where each of these chunks is signed separately. This segmentation should be done by e-doc software (because many messages can be easily utilised within data segment of a TCP packet). Note that such technique should be applied carefully. Example:**

*[The signer] [orders] [the transfer] [of US$] [$X_1$] [from account #] [$Y_1$] [to account #] [$Z_1$.]*

*[The signer] [orders] [the transfer] [of US$] [$X_2$] [from account #] [$Y_2$] [to account #] [$Z_2$.]*

**If each field in above messages is signed separately, variable fields become interchangeable. To prevent this, one possibility is to include a time-stamp in each signature (distinguished structuring).**

[2] This context dependency level should not be mixed with the level of practicable realisation as described in [Dob96a].

### *3.7. On-line services orientation*

**The more critical a transaction is, the more "on-line oriented" it should be. A transaction should be validated by the recipient shortly after the original document has been signed. Every collision search requires some time and thus reducing the gap between the time of the creation and the time of the validation reduces the risk of such attacks. One might oppose claiming that the purpose of the digital signature is to ensure non-repudiation generally and not just for a short time-interval after signing. But specific limitations are reality for digital signatures, e.g. the recipient can not be ensured about the e-doc non-repudiation properties as long as next CRL [X.509] is not issued (which enables the recipient to verify the corresponding public key). It is a simple fact that digital signatures aren't full equivalents to hand written signatures.**

### *3.8. Confidential envelope*

**Scrambling the whole e-doc along with signature using reversible cryptographic techniques could solve the problem of collisions. It would transform the problem from the area of hashing to the area of symmetric/asymmetric cryptography. Asymmetric approach (using a secret key) is acceptable only for short documents (it is a good practice to have one private key for the confidentiality and the second one for the authentication). The other solution is to use a symmetric session key. The task of the session-key exchange is provided by properly designed cryptographic protocols [Aba94], positioned at lower layers in the communication stack. Anyway, confidential envelope technique could be useful for the solving collision problems - it also ensures high security (confidentiality) for sensitive e-docs[3].**

### *3.9. Hash functions' output range*

**There is nothing revolutionary, but still worth to point out - the output range of possible hashed values should be as big as possible. This well-known fact should be taken into account when designing e-doc applications.**

### *3.10. Exclusion of hashing*

**The most conservative approach is to exclude hashing when signing the document, if acceptable. There is no formal bottom line with regard to computational infeasibility neither for idealised one way hash functions, nor for practicable implementations. However, the consequences of omitting hashing should be carefully studied. If nothing else, digital signatures don't have "normalised" values any more. This, on the other hand, complicates standardisation efforts, requires more storage space, etc. Moreover, using RSA [Riv78], which is the most popular algorithm to produce digital signatures, one must not use it without hashing to avoid homomorphic attacks. Thus for a general use, exclusion of hashing can hardly be considered a solution.**

## 4. Conclusion

**One-way hash functions are essential for provision of the security when using e-docs in electronic commerce. However, some widely used implementations (MD4 and MD5) have been shown to possess weaknesses that can be exploited to subvert digital signatures. Taking into account also the lack of formal proofs for one-way hash functions, it could be beneficial to design e-docs accordingly and to pay attention to the context of their usage. This is especially true for highly critical e-docs for financial transactions.**

**The paper presents a framework for minimisation of such risks with considering proper design of e-docs and the context of their usage. It is based on basic principles behind attacks on one-way hash functions. It shows that it is beneficial to treat e-doc exchange applications and security**

[3] As far as it is known to the author, this technique has been also proposed in the latest draft of S/MIME Implementation Guide (see http://www.rsa.com/S-MIME).

**mechanisms as a whole. However, guidelines in this paper are subject to a well-known trade-off paradigm -- to achieve better security, larger investment in time for the e-doc design is required.**

**Further work should be oriented toward more detailed analysis of e-docs on a sub-field level, i.e. their encoding. It might be also beneficial if standardisation efforts would put more pressure on the e-doc structuring, as almost no attention has been paid so far to minimise such risks (in EDI, for example, security as such is completely left to other systems). Changing this position would make electronic transactions more secure and put less tight requirements on one-way hash functions.**

## 5. References


**[Aba94]** **Abadi M., Needham R.M.,** *Prudent Engineering Practice for Cryptographic Protocols*, **DEC SRC Research Report 125, June 1994.**

**[ANSI81]** **American National Standards Institute,** *Data Encryption Algorithm*, **ANSI X3.92, New York, 1981.**

**[Bell95]** **Bellare M., Garay J., Hauser R., Herzberg A., Krawczyk H., Steiner M., Tsudik, G. Waidner M.,** *iKP - a family of secure payment protocols*, **Proc. of first USENIX Workshop on Electronic Commerce, New York, July 1995.**

**[Boer94]** **Boer den B., Bosselares A.,** *Collisions for the compress function of MD5*, **Advances in Cryptology, Proc. of Eurocrypt 93, LNCS 765, Springer Verlag 1994, pp. 293-304.**

**[Diff76]** **Diffie W., Hellman M.E.,** *New directions in Cryptography*, **IEEE Transactions on Information Theory, Vol. 22. No. 6, November 1976, pp. 644-654.**

**[Dob96a]** **Dobbertin H.,** *Cryptanalysis of MD4*, **Fast SW Encryption, Lecture Notes in Computer Science, Vol, 1039, Springer Verlag, 1996.**

**[Dob96b]** **Dobbertin H.,** *The Status of MD5 After a Recent Attack*, **CryptoBytes, RSA Labs, Vol. 2, No. 3, 1996, pp 1-6.**

**[EDI93]** **United Nations Economic Commission for Europe,** *Electronic Data Interchange for Administration, Commerce and Transport*, **Syntax Rules, ISO 9735, March 1993.**

**[NIST93]** **National Institute of Standards and Technology,** *Secure Hash Standard*, **NIST FIPS PUB 180, Dept. of Commerce, May 1993.**

**[Gal91]** **Galvin J.M., McCloghrie K., Davin J.R.,** *Secure Managemebt of SNMP networks*, **Integrated Network Management, II, North Holland, 1991, pp. 703-714.**

**[ISO88]** **International Standard Organization,** *Information Processing Systems, OSI Reference Model - Security Architecture*, **ISO 7498-2, July 1988.**

**[ITU96]** **ITU-T Q15/7,** *Final Text of Draft Amendments DAM 4 to ISO/IEC 9594-2, DAM 2 to ISO/IEC 9594-6, DAM 1 to ISO/IEC 9594-7, and DAM 1 to ISO/IEC 9594-8 on Certificate Extensions*, **Geneva, April 1996**

**[Kal92]** **Kaliski B.S.,** *The MD2 Message Digest Algorithm*, **RFC 1319, RSA Data Security Inc. 1992.**

**[Mast96]** **MasterCard & VISA,** *Secure Electronic Transactions*, **Draft specification at http://www.mastercard.com/set, July 1996.**

**[Pre95]** **Preneel B., Van Oorschot P.C.,** *MDx-MAC and Building Fast MACs from Hash Functions*, **Proc. of Crypto 95, Springer Verlag, Aug. 1995.**

**[Rob96]** **Robshaw M.J.B.,** *On Recent Results fir MD2, MD4 and MD5, RSA Labs Bulletin*, **Vol X, No. 4, Nov. 1996.**

**[Rog95]** **Rogier N., Chauvaud P.,** *The ompression of MD2 is not collision free*, **Proc. of Selected Areas in Cryptography 95, Ottawa 1995.**

**[Pri89]** **Davies D.W. and Price W.L.,** *Security for Computer Networks*, **John Willey & Sons, New York 1989.**

**[Riv78]** **Rivest R.L., Shamir A., Adleman L.,** *A Method for Obtaining Digital Signatures and Public Key Cryptosystems*, **Comm. ACM, Vol. 21, No. 2, February 1978, pp. 120-126.**

**[Riv90]** **Rivest R.,** *The MD4 Message Digest Algorithm*, **Proc. of Crypto 90, Springer**

**Verlag, August 1990.**

**[Riv92]** **Rivest R.,** *The MD5 Message Digest Algorithm*, **RFC 1321, IETF, April 1992.**

**[Tsu92]** **Tsudik G.,** *Message Authentication with One-Way Hash Functions*, **ACM Computer Communications Review, Vol. 22, No. 5, 1992, pp. 29-38.**

**[X.12]** **ANSI ASC,** *American National Standard for electronic business interchange*, **X12 series Standards.**

**[X12-840]** **ANSI ASC,** *Request for Quatation*, **Transaction Set 840, Release 003040FED1, August 1994.**

**[X.509]** **ITU-T,** *The Directory: Authentication Framework*, **Recommendation X.509(E), Geneva, 1993.**

**[X12-850]** **ANSI ASC,** *Purchase Order*, **Transaction Set 850, Release 003040FED1, August 1994.**

## 6. Appendix

**The purpose of this short appendix is to assure a common understanding of two kinds of collisions that were referenced in the paper.**

### *6.1. Birthday paradox collision*

**Suppose there is a universal (non-finite) set** $A$ **with elements, which are mapped to elements of a finite set** $B$**, which are integers in the range from** $1$ **to** $n$**. For each element in** $A$ **the probability of mapping to a certain element in** $B$ **is equally likely. Put another way, hashed value is a random integer variable with a uniform distribution between** $1$ **and** $n$**.**

**After randomly choosing** $m$ **elements from** $A$**, what is the probability** $p$ **that there is a collision? When choosing the first element, the probability** $p_{cf}$ **of "not-already-chosen" element is** $1$ **(subscript** $cf$ **stands for collision free). When choosing the second element, the probability of "not-already-chosen" date is** $((n-1)/n)$**. Going on we obtain** $((n-2)/n)$ **etc., and the probability of non-collision is given by the following equation (subscript** $c$ **denotes collisions, while subscript** $cf$ **denotes non-collision):**

$$p_{cf}(n, m) = (n/n)_1 ((n-1)/n)_2 \ldots ((n-m+1)/n)_m$$

**Thus the probability for a collision is**

$$p_c(n,m) = 1 - p_{cf}(n,m) = 1 - (n/n)_1 ((n-1)/n)_2 \ldots ((n-m+1)/n)_m$$
$$p_c(n,m) = 1 - p_{cf}(n, m) = 1 - n!/n^m(n-m)!$$

### *6.2. Fixed-date paradox collision*

**Let' s take the same assumptions as stated above. Opposite to the birthday paradox, let' s fix a particular element in** $A$ **in advance and choose additional elements until a collision in** $B$ **occurs. Using analogous reasoning as above, the probability for a collision is**

$$p_c(n,m) = 1 - p_{cf}(n,m) = 1 - (n-1)^m/n^m$$