

Internet Draft
draft-ietf-ltans-arch-00.pdf
November 2003
Expires May 2004

Libor Dostalek
PVT
Marta Vohnoutova
PVT

Long-term Archive Architecture

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of [RFC-2026].

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced or made obsolete by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Copyright Notice

Copyright (C) The Internet Society (2003). All Rights Reserved.

Abstract

At present, the mankind creates digital documents with digital signatures in a mass scale. This draft is focused on the issue of archiving of digitally signed documents.

First of all, it is necessary to summarize who creates the documents with which the documents are exchanged and what are their basic features. With the help of such analyses it is possible to define a technical specification of an intended archive of digital documents.

The goal of this document is to summarize particular protocols which are necessary to be able to build the archive of digitally signed documents.

This is the first version of the dokument. Some problems remain open in this version. These open problems are listed in the Conclusion chapter. The reason is that it seems to be necessary to make some minor changes into some already existed or suggested protocols to interlock mutually. If the protocol contents correspond we will be able to create real archives of digitally signed documents based on them.

We are fully open to other co-authors to create next versions together and solve all the already unresolved problems. We wish to create gradually the cook book of digital archive architecture. This material could be a starting point.

Table of Content

1	INTRODUCTION.....	3
1.1	LEGISLATION VIEW.....	3
1.2	LONG TERM SIGNATURE	3
1.3	DISCUSSION WITH PETER SYLVESTER	4
1.3.1	<i>Trustworthy archives.....</i>	5
1.3.2	<i>Sylvester's cone</i>	6
1.4	TRUSTWORTHY ARCHIVES.....	6
1.5	THE VIEW OF REAL ARCHIVISTS.....	7
1.6	MIGRATION AND EMULATION.....	8
1.7	MIGRATION OR EMULATION	9
1.8	FORMATS OF DIGITAL DOCUMENTS	9
1.9	I DIGITALLY SIGN ONLY WHAT I CAN SEE.....	9
1.10	FORMATS OF SUPPORTED DIGITAL SIGNATURE	10
2	EXCHANGE OF DOCUMENTS	10
3	WEAKENING OF USED ALGORITHMS	11
3.1	CERTIFICATE EXPIRATION	11
4	ARCHIVED DOCUMENTS	12
4.1	DIGITAL DOCUMENTS	12
4.1.1	<i>Documents without digital signature and time stamps.....</i>	12
4.1.2	<i>Documents with digital signatures and time stamps</i>	12
4.1.3	<i>Scanned documents</i>	13
4.2	ORDERING OF DOCUMENTS IN ARCHIVE	13
4.3	EXPORT OF DOCUMENTS AND THEIR IMPORT INTO UPPER LEVEL ARCHIVE	14
4.4	METADATA	14
5	MAIN FUNCTIONS OF ARCHIVE.....	15
5.1	CONFIRMATION	16
5.2	STORING OF DOCUMENT INTO ARCHIVE.....	17
5.3	SEARCHING FOR DOCUMENT IN ARCHIVE.....	18
5.4	SHREDDING OF DOCUMENT	18
5.5	RENEWAL OF DOCUMENT	19
6	STANDARD REQUIREMENTS.....	19
6.1	DATA FORMAT.....	19
6.2	MESSAGE FORMAT	20
6.2.1	<i>Archived-data Content Type.....</i>	21
6.2.2	<i>Useful attributes.....</i>	23
6.3	TYPE OF INKLINGS.....	24
6.4	ARCHIVE TIME-STAMPS SYNTAX [ATS].....	24
6.4.1	<i>Objections to ATS.....</i>	25
6.5	ENHANCED TIME STAMPS	25
6.6	RECEIPTS ISSUED BY ARCHIVE.....	26
6.7	AUTHENTICATION OF CLIENT ENTERING ARCHIVE.....	26
6.8	TAP	26
6.8.1	<i>Present TAP protocol draft.....</i>	26
6.8.2	<i>Objections to TAP protocol.....</i>	28
7	CONCLUSION.....	30
8	ACKNOWLEDGEMENTS.....	31
9	REFERENCES.....	31
10	AUTHORS' ADDRESSES	31

1 Introduction

1.1 *Legislation view*

Almost all European countries have their own national legislations which are less or more derived from the European Union directives. Particularly for me, it was very interesting to realize that two totally different areas are involved in this topic:

- The area solving the digital signature act
- The area which is involved in archiving of documents

Authors of various laws about the digital signature act mostly do not take care about archiving. E.g. according to Middle-European laws (which are derived from the "old good" Austro-Hungarian Realm law) almost everybody who produces some documents should have his own "archival and cancellation order". In this archival and cancellation order, there is specified the way how documents should be archived and which documents are of permanent value and thus they should be put into the next archive after they are cancelled.

Because it is not clear how to archive digitally signed documents for long term (permanently), then (theoretically) digitally signed documents of permanent value cannot even be created.

But the practice is different. Digitally signed documents of permanent value are created even nowadays. According to my opinion, an example of such a document is e.g. a certificate of a certification authority, which issues qualified certificates [RFC-3039]. How we archive it now? The certificate is printed out in the PEM format and is published in the order of law. The archive is used to archive this order of law (in a paper form). We have had to enter this printed out certificate back into a computer several times, this is a many hours work even for an experienced "digital archaeologist". If the quality of printing is not so good, it is difficult to distinguish capital O (ou) from 0 (zero) etc.

And we still did not mention users' certificates. Archivists do not take care for them because passports or Identity Cards they also do not archive. But this user's certificate could be e.g. the certificate of the president of your republic! Fortunately up to now, presidents have used their certificates to play with (after he signed digital signature act) and they have not already signed any official document with it. What, my goodness, we would do with such a document!!!

Archivists know it, they call our attention to this problem and they are right that they do not want to accept any digital document into their archives. Even though they have plenty of such documents there and some of them are stored in media that I had never heard about before, even if I work in the IT more than 25years.

1.2 *Long term signature*

At the very beginning we thought that the problem was not so complicated. We knew the standard [RFC-3126] (Electronic Signature Formats for long term electronic signatures) which was completed by the ETSI and issued as the ETSI TS 101 733 standard. It seemed to be simple for us. It is enough to add a time stamp derived from a digital signature to the digital signature. This will prolong the digital signature validity to the time stamp validity. Before the TSA certificate expires, we can simply add a next archive time stamp to the digital signature using the new TSA certificate for its creation, and the validity of the digital signature is prolonged again. We can repeat these steps up to now.

The time stamp derived from the digital signature is the evidence that the document existed before the given time, i.e. a potential hacker could not have enough time to find out the particular private key to the public key which is in the certificate.

The main fault (disadvantage) of [RFC-3126] is its dubious usage in case of documents which contain more parallel digital signatures. Of course it is possible to timestamp each of digital signature independently, but this method does not prevent the following potential attack: a hacker removes some of digital signatures including all their relevant time stamps (after all the base of such an idea you can see in the TAP protocol – see[TAP]). Also the "inflating" of each document with trustworthy anchors is probably in vain in archives, because the majority of documents from the same source of documents will use the same trustworthy anchors for some time. As for me, it is enough to store the trustworthy anchors in the metadata which are kept for the particular data source.

The archive of digitally signed documents according to the [RFC-3126] would be "a data warehouse" which appends a document according to the [RFC-3126] to the ES-T, ES-C, and ES-X up to ES-A form. Next, it will ensure that the archival time stamps (in the ES-A form) are added to the documents regularly at least before the particular TSA certificate expiration.

Please give your attention to this, because, unlike us, some people think that it is not necessary to timestamp the document before the particular TSA certificate expires. I.e. it is not necessary the archival time stamps would overlap.

If the man dies, his certificate is revoked, but the documents signed with his certificate before it was revoked, are valid. You can take care of this with the help of the digital signature policy, you can write there that a digital signature is complete only when the document is in the ES-T form, i.e. the time stamp from the digital signature is added to the document. Consequently, you can prolong the digital signature validity with the help of time stamps up to now.

1.3 Discussion with Peter Sylvester

In spring 2003 we visited Mr. Peter Sylvester in Paris. The discussion with him influenced us and forced us to think over some of our opinions again. He considers the digital signature as a short time matter. He considers the mechanism "Long Term Signature" acc. to [RFC-3126] and taken by ETSI TS 101 733 a failure. I argued with him that this is a standard. He gave me an example of many standards with the similar doom.

Many of his arguments persuaded us. (Till the time, we visited the real archivists who completed this idea with the fact that the archive cannot be absolutely trustworthy). Their approach is different they assert that if a particular document is valid, it means that the trustworthy party declares it is valid. E.g. a notary takes care of a testament, but the probability that the signature in the testament is not genuine (that would mean that the notary is a swindler) is really very low. It is important the document is digitally signed and then stored to a trustworthy archive. When the document is picked up from this archive after some time, it is not necessary to verify the validity of digital signatures, because if they had not been valid, the document would have never been stored into the trustworthy archive.

The [RFC-3126] has an idea that I will prove the document validity if I steadily prolong the validity of old certificates, OCSP responses or attribute certificates (which give the evidence that the particular person was entitled to sign it) by adding another and another time stamps to them.

The parallel from the real life is, as if we need to prove the validity of old documents, we would have to bring some old, long time ago expired Identity Cards with us and prove that they were valid that given time.

We were surprised that I could not see such obvious things myself. I was sitting and imagining the digital parallel of the document validity verification according to the [RFC-3126] in a real life. As an example I took the "founding document" of the Charles University in Prague. This is the document we still believe it. If I want to verify this document acc. to the [RFC-3126], I should:

- Verify that the king Charles the Fourth was entitled to sign the "founding document" of the Charles University. It means he should have to have his attribute certificate, in which should have been written that he was a king and he was entitled to rule the kingdom.
- Verify that he had even his signing certificate (i.e. the digital parallel of the Identity Card).
- The signed document would have been appended into the ES-C form. Before some of the certificates required for the digital signature verification expired, the digital signature should have been time stamped. But this time stamp has its own digital signature (in this case the one of the TSA) which is also vanishing, then also this digital signature should be time stamped from time to time. If this has been done with e.g. six-month periodicity, nowadays, we would have more than 1300 stamps (husks) appended to this particular document!!!

If I want to verify my university diploma, I would have to verify the "founding document" (to prove that the Charles university exists) and logically also to prove that the rector was legally elected. That e.g. some of his predecessors who were beheaded (e.g. Johannes Jessenius, beheaded in the Oldtown Square in Prague in 1621) is not still the valid rector of the University etc.

Even if all the verifications are successful what this would mean in practice? NOTHING. What should I do now with the certificate and the attribute certificate of the Charles the Fourth? Even if I would formally verify its validity using the RSA, in reality it would mean nothing. During centuries, the regimes have changed many times. And has the new regime accepted the acts done by the previous regimes?

NO,NO this mechanism is not used (and cannot be used) in practice. The signature of a document is only a ceremony when the entitled person signs something witnessed by others persons. This proves the validity of the document only for a short term. If the document is valid for a long term, it means that the state will accept this document into its "trustworthy" royal/imperial/royal-imperial/state archive. Notaries follow the same principle. In this case the notary accepts the document into his archive. The similar situation is in case of companies. We can at more be given by some confirmation of the document validity (e.g. by the DV-certificate – see [RFC-3029]) when picking up a document from an archive.

1.3.1 Trustworthy archives

Any document should be verified and checked before storing it into the archive. It can be e.g. appended into the ES-C form. It can be time stamped from time to time, but it is not possible to build its validity only on these stamps.

The trustworthy storage should have three basic features:

- It should guarantee that the archived document will not be lost or modified.
- It should guarantee that no unauthorized person will be able to access the document.
- It should guarantee the sequence of the documents stored into the archive. It means if someone wants to cast doubt on the date and time when the document was stored, he should have to doubt also the date and time of documents stored to the archive before and after this particular document. It is more complicated and other persons witnessed the storage process. The sequence of stored documents could be done with the help of e.g. linking hashes.

We would like to add the fourth important feature:

- The trustworthy storage should archive only genuine documents. The real meaning of the word "genuine" must be determined in the policy of the archive.

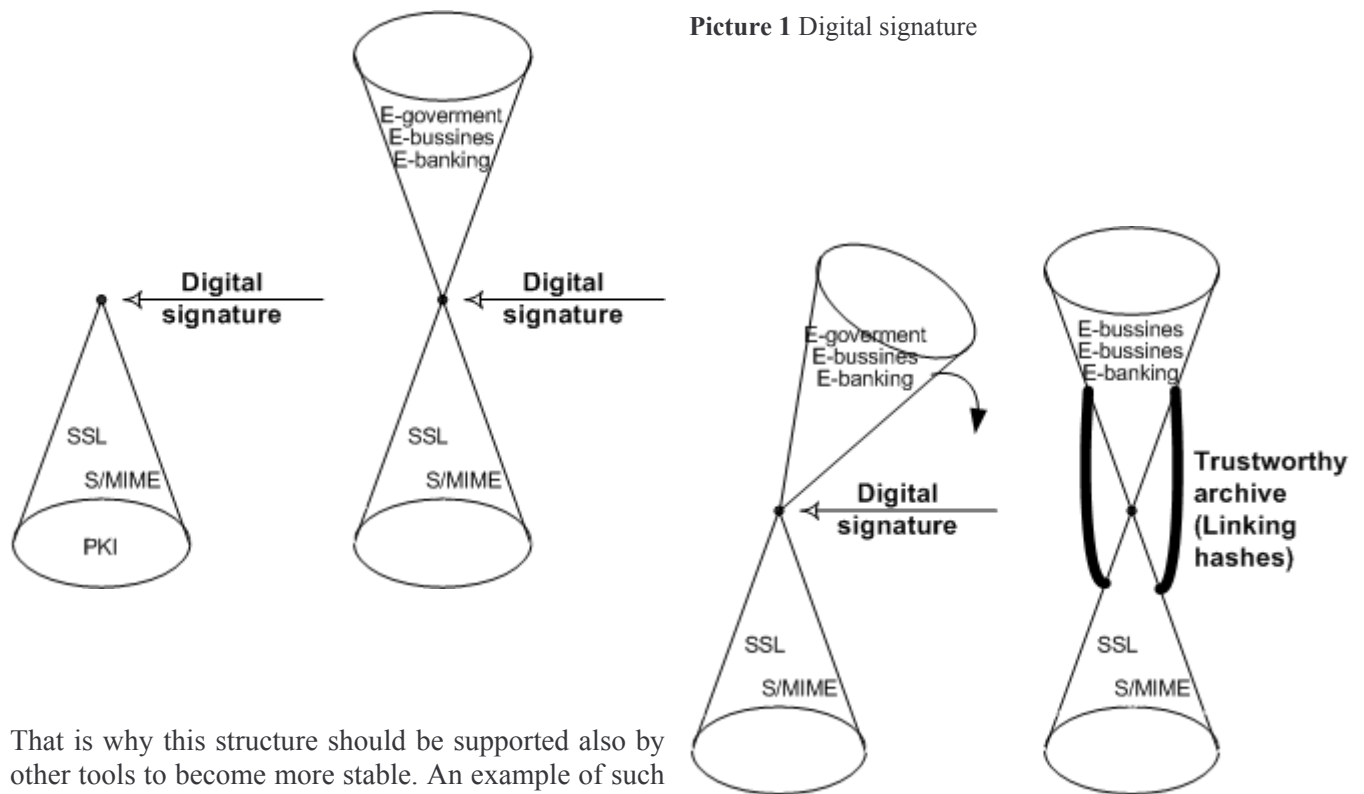
1.3.2 Sylvester's cone

To give us some evidence that we overestimate the importance of digital signature, Peter Sylvester drew a very interesting picture: a cone with a digital signature on its peak.

He explained his drawing: there are various protocols above the PKI and on the very top there is a digital signature. The dream that this digital signature could ones replace a handwritten signature is just an illusion which cannot come true, that is why it is on the unreachable peak.

And all the applications called "e-something" are built above this peak.

Everything is built on the RSA or similar theories. And only a little shaking would cause that this structure becomes unstable or falls down:



Picture 1 Digital signature

Picture 2 Sylvester's cone

1.4 Trustworthy archives

Trustworthy archive is an authority into which we give our digital documents and we trust it that it does not change them, lose them or does not allow any unauthorized person to access them. If the trustworthy archive verifies the digital signatures of the document before archiving it, the digital signatures should remain valid also after picking up the document from the archive. The archive confirms the user the acceptance and archiving of his document. If the document is picked up from the archive, the archive gives to the user a confirmation of the digital signatures validity and authenticity. Is this idea right or is it only the idea of IT people?

1.5 *The view of real archivists*

At present, IT people are involved in many non-technical problems. Their knowledge is logically limited but they think to understand everything (including archives) and to be able to solve problems that more informed people have not been able to solve for years. Up to now we saw the archive from the point of view of IT people or maybe also from the point of view of lawyers who see the document only for a short time after its creation.

It should be a good idea to ask real archivists for their point of view, because they will decide about the acceptance of a document into an archive. If they cannot accept the digital document, they will force us to print it out and put it into the archive in a paper form.

We discussed this topic with archivists. This discussion was really very instructive for us. We begun that we would want to build trustworthy archives of digitally signed documents. The fat historian in front of me seems to be unable to move faster. But he responded very quickly and told me that he could not understand: "What does it mean the trustworthy archive?" He pretended he had never heard about it. (But he knew the wrong ideas of IT people about archives very well and he only prepared his own attack.)

I explained him my idea of the trustworthy archive (see the preceding chapter):

- Before archiving, the archives verify the validity of the document.
- Ensure that the document will not be lost in the archive
- When the document is picked up from the archive, the archive will confirm the validity of the document. The archive can do it because it already verified the document validity when it archived it. And logically because the archive is trustworthy thus the picked up document must be genuine.

He was prepared to crash me up: Archives are not involved in verification of documents validity before their acceptance. It is true that archivists work with documents in organizations where the documents are created. They can discover some forgeries when looking into documents before their archiving but it is really not the goal of their mission. To prove his assertion he told me: "Do you know what documents in our archive we are most proud of? The medieval forgeries!!!"

He also did not accept that archive should guarantee that no documents in archives would be lost. He had very good objections. "You know that archives are moved e.g. in war times. Because we measure many kilometers of documents in archives, it is not an exception that even a wagon of documents is lost. Or another example: Our archives contain the very complex archive of Gestapo. You cannot imagine how many previous Gestapo co-workers penetrated into the archive telling various pretexts to steal their compromising documents."

We should leave the term "trustworthy archives". No archive is absolutely trustworthy. There are employees in the archives and we know well that 90% of deceits around the World are made by employees.

Archive only stores a document. The document must contain evidences about its authenticity. But only a court is entitled to decide if the particular document is genuine or not.

Only a court can decide if a document is genuine or not. The picked up document must have so many evidences that it is genuine that the court has no doubt about the authenticity of the document. That is why the court often declares a genuine document not to be genuine.

I wrote into my notes: "The archive is not trustworthy". The peaceful historian looked at it, got red and shouted: "What are you writing? You have written that our world-known archive is not trustworthy!!!" I realized that the term "trustworthy archive" is the term of IT people; archivists know well that this

term is nonsense similar to e.g. perpetual motion. Both are impossible to solve but all the time some people are trying to do it.

Also the IT people should realize that people work even in trustworthy archive and people can fail. Even the distributed archive is not the solution. The archived documents cannot leave the state and in case of catastrophe could be also destroyed. But the most danger for trustworthy digital archives is software authors. A software error can damage the distributed archives as well because software errors are also distributed. Banking software authors could talk how their reliable "trustworthy" software doubled payment orders and after how many days such an error was discovered.

We simply cope with the fact that either Internet is not a secure environment, or archives are not fully trustworthy.

Archivists emphasize that we cannot overestimate so called "natural scientific verification of document authenticity" (from the archivists point of view, not only mathematics but also technical science is a natural science).

They can prove their assertion easily: "No medieval forgeries were discovered because of natural scientific survey of the document, in all cases the forgery was discovered as a result of historical survey." A historian found some data in the document, which could not be true in the alleged time. E.g. a building mentioned in the document was build 200 years later. Archivist added the known sentence: "The found source is from 1885 what was verified with radiocarbon method with accuracy plus minus 300years".

1.6 Migration and emulation

If we really want to archive digital documents for a long term, on the contrary of the paper documents there are some troubles with them. Everyone who works with digital documents for some time has already faced this. Formats of digital documents are not unique. Imagine that you want to view older document created in MS Word in some newer version of this program. If you succeed by chance, you will probably see "a bit different" document than it was before.

We have two basic methods to work with documents, which were created in the IT environment which is not used nowadays:

- Emulation. I.e. The old IT environment is emulated in the present IT environment. I faced several times when the IT moved to a new generation (e.g. moving from 8bit computers). And I faced the emulation in practice. It was newer enough funds and tools for emulators to be fully reliable.
- Migration of data from the format used in the old environment to the "more modern" format. This is probably more reliable method but we must be aware of the fact that always some historical information will be lost. And probably only the God can decide if such lost information were valuable or negligible. Physicists could talk us about the influence of "waving of butterfly wings" to the very long term predictions of moving of planets and stars, because negligible things which are cumulated during centuries can valuably influence behavior of big planets and stars. In any case we should save also the original document to be able to decide in future if migration was precise or if it distorted the document.

Theoretically, there are also other methods e.g. an encapsulation of a document together with its IT environment, but these methods are not so common.

I think, it is interesting that it is discussed what is better if the migration or the emulation and it is looked for some absolutely perfect method. But the reality is different. We have many formats of

digital documents: music recordings, films, executable programs, databases, web servers, our digitally signed and many others.

I think that various formats need various methods. E.g. it is quite suitable to archive films. Historians can easily prove the historical authenticity of films because they know how buildings and countryside looked that time.

1.7 Migration or emulation

It is impossible to change any bit in digitally signed documents because their digital signatures would become invalid. But migration changes the document!

If we want to leave the document authentic, I cannot change it. It means that unlike the others the digitally signed documents cannot be migrated or emulated.

They must be created in the form which is not necessary to migrate or emulate. This format exists - it is the plain text with the UTF-8 encoding.

1.8 Formats of digital documents

I think that in case of permanent value document it is necessary to use the plain text format. These documents must be stored this way until other formats, which are not necessary to migrate, are defined.

This could be unpleasant information for users of MS Word and other editors, but these editors were created for a different purpose: they are used for creating of printed documents and not for digital documents suitable for permanent archiving. These editors are determined for different purposes. Remember how many information could be hidden in document revisions or how the document view is different when using another version of editor to view it. If we want to archive documents for long term it is better to make them in the plain text format than to be forced to print them out before archiving them.

The digitally signed document is an exceptional kind of a digital document; it wants to give people the evidence of the document validity. This evidence is a digital signature and because of this such a document cannot be migrated. And we already know that the emulation is not reliable. That is why such a document must be archived in a format which is not necessary either to emulate or to migrate.

Another possibility is to migrate the document without digital signature.

If we migrate a document and leave its digital signature as it is, we would destroy the evidence of the document validity for future research workers. We could compare this method with a medieval document with wax seal: we will retype this document in MS Word and to research workers who will survey the document validity we will give a sack with the wax melted from the original medieval seal.

1.9 I digitally sign only what I can see

In their laws of a digital signature of many countries, there is a condition that a user digitally signs only exactly what he can see. It means a user should not digitally sign a document which could be explained in a different way - e.g. against his favor.

This is the second reason not to digitally sign documents created in editors determined for document printing (e.g. MS Word). Even e.g. a document in the XML format created in the MS Word editor is so complicated, that even quite experienced user cannot be sure about the precise content of such a

document. View the source text of this document to a common user and ask him if it is the exact content of his text!

If we start to digitally sign documents in the plain text format, users will not be satisfied. I presume that this dissatisfaction will lead to specifications of new formats of digital documents suitable for permanent archiving.

I realize that creation of documents in plain text is not convenient. But if we go back to the time when we created web servers with the “vi” editor and used five tag types, we were able to work with such texts and they were readable for common users. It would be quite simple to specify such tags in metadata.

1.10 *Formats of supported digital signature*

We have two formats of digital signature in practice (I do not consider EDI format and other not so common formats). I mean the digital signature in the CMS format (PKCS#7) and the XML (see [RFC-3369] and [RFC-3275]). Both formats have developed independently, but from the archiving point of view we must solve archiving of both. The XML format does not contain some features of the CMS format because its development stagnates. Would it mean that we should take the document with the XML signature and put them as a data into a CMS message to have the required archival format? Is this really our goal?

2 Exchange of documents

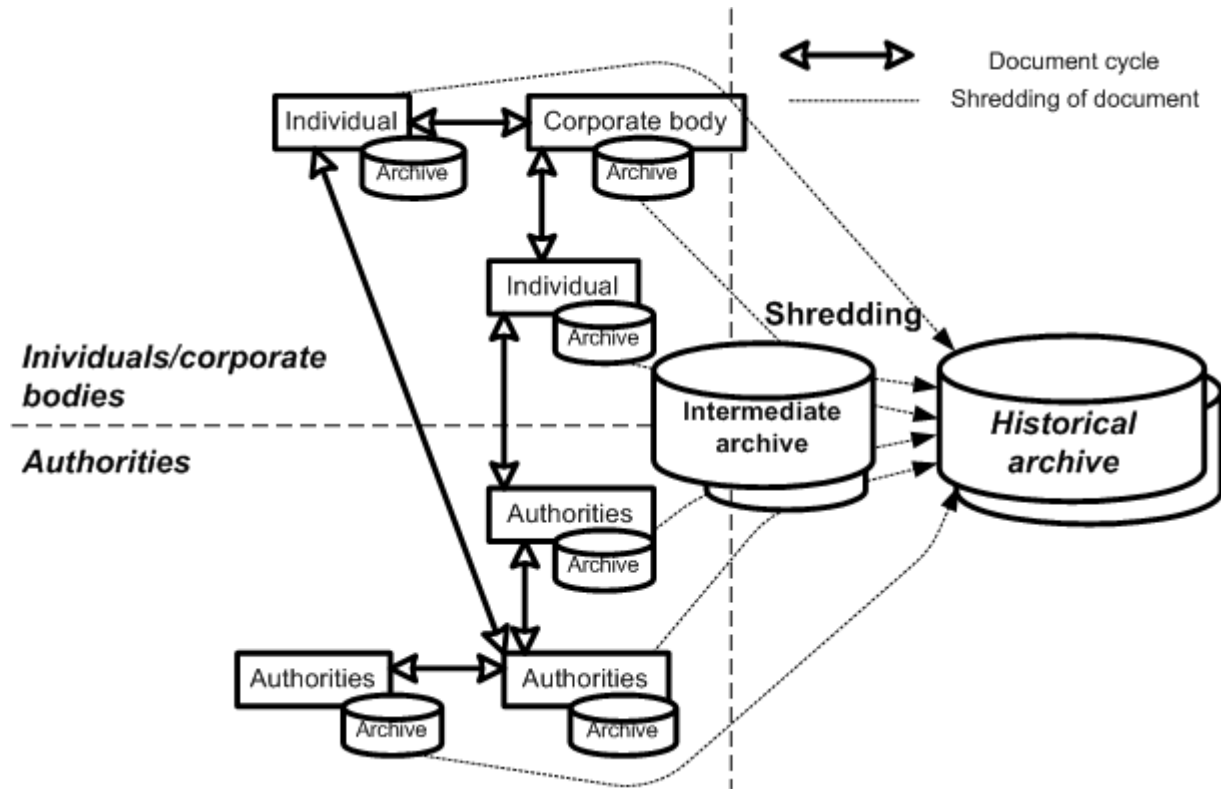
Documents are exchanged between (see picture 3):

- Particular persons (either individuals or corporate bodies). The typical examples of such documents exchanging between persons are invoices.
- Particular persons and authorities (i.e. establishment or public institutions). The typical examples of such documents exchanging between persons and authorities are income-tax returns, planning permissions etc.
- Authorities. The typical examples of such documents exchanging between authorities are various reports sent from lower-level authorities to upper-level authorities.

Every person or office is obliged to archive the documents specified in its archival and canceling order for the defined time. After this time, the document is either destroyed or put into the historical archive.

There can be one or more intermediate archives inserted between archive of an individual/corporate body archive and a historical archive. The typical example of usage of such an intermediate archive is the situation when an individual/corporate body creates classified documents (secret, top-secret etc.). The documents are archived in the intermediate archive till their classification (secrecy) is over. That is why the historical archive does not classify documents (they are free).

We can also have more types of „historical“ archives. On one hand central archives can be established by various central authorities (e.g. agricultural archive, film archive etc.). On the other hand if the documents are owned by individuals, the state cannot take such documents into the historical archive. But it can order them to run the private historical archive of parameters comparable with the one run by the state itself.



Picture 3 Document cycle from source to historical archive

3 Weakening of used algorithms

[Snatch of the ATS draft]

In many application areas of electronic data exchange a non-repudiation proof of existence of data has to be possible over long periods of time. An important example is signed documents, which sometimes have to be archived conclusively over 30 years or more. During the archiving period hash algorithms and public key algorithms or their parameters can get weak or certificates can become invalid. To avoid that signatures lose their probative force it has to be provable that the data already existed before such a critical event.

This can be done by timely generating Archive Time-Stamps for these data and by renewal of these Archive Time-Stamps during archiving period.

3.1 Certificate expiration

The main problem of digitally signed documents is that they must be valid even the relevant certificates necessary for the digital signature verification are already expired. The problem could be solved if we add another inkling which could prove that the document was digitally signed in the time when the relevant certificates were valid. Such inkling is also evidence about the moment when the document was signed - what could be also important for the validity of the document as a whole.

The inkling seems to prolong the strength (I purposely did not use the word "validity") of digital signature. The appending of this inkling is called renewal of the document.

We know two types of inklings till now:

- Inklings based on the time stamp from the digital signature. The [RFC-3126] (Electronic Signature Formats for long term electronic signatures) deals with this technology. But the strength of this inkling is also time-limited because the time stamp is also digitally signed data

structure which is also verified with the help of certificate (the TSA certificate in this case). That is why the [RFC-3126] deals with the regular renewal of documents.

- Inklings based on linking hashes. The draft Archive Time-Stamps Syntax (ATS) deals with this technology. Also this technology is time-limited which depends on the strength of the algorithm used for the hash calculation. But this technology seems to be less time-limited and could be valid for considerably longer time compared with time stamps.

If digital documents are not digitally signed they can be amended (at the latest – at the moment of its archiving) with the time stamp of the documents. Keep in mind that the time stamp is also digitally signed data structure and thus must be renewed as well.

4 Archived documents

4.1 Digital documents

The main thing where digital documents differ from paper (parchment etc.) is that they are stored independently from data medium. It means that in case of digital documents we archive only information and not the data medium. The data media archiving is possible only if we archive for relatively short time (units of years). The main reason is not only short durability of data medias but mainly often changing formats of data storing into data medias (e.g. file structure systems).

During copying of data from one data media to another data media we often migrate the format of data storing. But the migration of particular documents is not so often.

When copying data between data medias (renewing of data medias) some data media features could be lost e.g. the fact that the data media is WORM.

Digital documents can be divided into:

- Without digital signature and time stamps
- Digitally signed documents including those with the only time stamp
- Scanned documents

4.1.1 Documents without digital signature and time stamps

The documents contain historical information. Either digital signature or accurate time of their creation is not significant. The archive can amend simple or extended time stamp to these documents as an inkling of their existence in a concrete time. The disadvantage of such append is the fact that the time stamps and signatures must be renewed not to weaken during time.

4.1.2 Documents with digital signatures and time stamps

If we want to keep the strength of digital signatures and time stamps we must renew them. One of the reasons is the fact that the document can be picked up from the archive because of a legal proceeding. To be accepted in the legal proceeding the court must verify the authenticity of the document first. To be able to verify the authenticity of the document the court need to have a maximal number of inklings which can prove the authenticity of the document

The fact, that the document can be picked up from the archive and can be used as a burden of proof, is time-dependent. In case of some documents we can do it only several years, in case of others we can do it for a very long time. For example the authenticity of an invoice is important only for the time when the person can be accused of a fraud while the documents about the acquisition or expropriation of a property can be important even after hundreds years.

It means that to renew digital signatures and time stamps not to lose their strength makes sense only for the time when they can be used as valid documents. It is possible to archive then after this time but

only as a historical heritage – in other words for the reasons of a historical survey. I think in that case it is no need to renew digital signatures and time stamps.

It means that in case of digitally signed or time stamped documents we must define in an archiving and cancellation order not only the time of their archiving but also the time of their renewal.

4.1.3 Scanned documents

This is a special case of digital documents which were created as a copy of original paper documents. Archives scan a maximal amount of written documents to minimize the manipulation with original documents. I think that researchers would appreciate a DV-certificate as a verification that the document was given to him by a concrete archive.

Scanned documents can be but do not need to be amended with:

- Time stamp as an inkling that the document exists before a time written in a time stamp.
- Digital signature of a notary who verifies the authenticity of a document. In some countries (e.g. in Austria) notary offices create this way digital backups of original documents created by notaries. In this case a notary puts his handwritten signature into the original document and digitally signs the scanned copy of the document.

If we digitally sign or timestamp the scanned document we must handle with this document as with the digitally signed one. This means that we must renew signatures and time stamps as we have described in the above article. It is also possible that the time of renewals is set to zero i.e. the document is signed/timestamped once and no renewal is required.

4.2 Ordering of documents in archive

Documents must be stored in an archive in such ordering (organization) to be able to find out them later. In other way the document could be lost in an archive.

It is important that digital documents arise or are processed in various information systems (as DMS etc.). Documents stored in the archive must be searchable even after the end of the information system. It would be wrong if the document, entering the archive, have obtained even unique but unpredictable identification. In such case it would be impossible to find out the document without the help of the original information system in other way than to go sequentially through the whole archive.

The different ordering will be used by a person, an authority or even the historical archive:

- Person will use sorting according to **agenda** (incoming invoices – i.e. a document about the VAT, wages, employees,...). The documents can be sorted by e.g. date into each **agenda**.
- Authorities and offices can have two archives from the ordering point of view:
 - Office is also a corporate body (person), i.e. it has the invoice archive, archive of employees etc.
 - Office communicates with the public (persons), i.e. it must have an archive of received and sent documents. These documents are usually sequentially numbered within an actual calendar year. The order follows how they were received or sent.
- Historical archive must have the basic sorting according to the source from which the document was given into the archive. Since the source gives to the archive only a little portion of its documents, it is possible that particular documents are sorted only by the date of creation within the source ordering.

The main objection could be that the majority of documents is archived only for a short time or at most for a limited time. I.e. it is sufficient to keep the documents in the information system of the company (or better in the backups of the IS).

But the system must be prepared to give a part of its documents into the historical archive. In addition the company changes its IS or at least the version of its IS. It is obvious that the export of long-term archived documents into the archive is always the profit for the company. We can solve some critical or disastrous situations: person's death, company failure or the authority cancellation. When the archiving of important documents is necessary before the information system is switched off.

4.3 *Export of documents and their import into upper level archive*

The document can be exported mechanically, but if it is exported into the upper level archive the following must be fulfilled:

- The new identification of the document is created (the previous identification can be only an additional information).
- The historical information about the document stored in the document metadata are verified, amended and corrected.
- The information about the searching for the document within the archive stored in the document metadata (e.g. keywords) are verified, amended and corrected.
- If it is necessary to renew the signed/time stamped document also in the upper level archive, this renewal should be made again from the beginning i.e. in the ATS terminology: new Archive Time-Stamp Chain.

4.4 *Metadata*

Data about data. They must be at least so complex to be able to work with the document even if the information system, which had worked with the document, was cancelled. The metadata must contain not only the historical information about the document but also the information to search for the document in the archive.

Each document source has its own metadata about itself as a document source:

- Name of the document source
- Identification of the document source (person, office, application,...)
- Historical data about the document source
- Specification of the ordering of documents from this document source
- For digital documents:
 - o Information about document formats.
 - o Information about migrations/emulations
 - o Data for verification of digitally signed documents (trustworthy anchors, CRL, OCSP, ...)

At least the following metadata are kept for any particular document:

- Identification of the document source
- Identification of the document (reference number, folder or box identification + identification of the concrete document etc.)
- Date and time

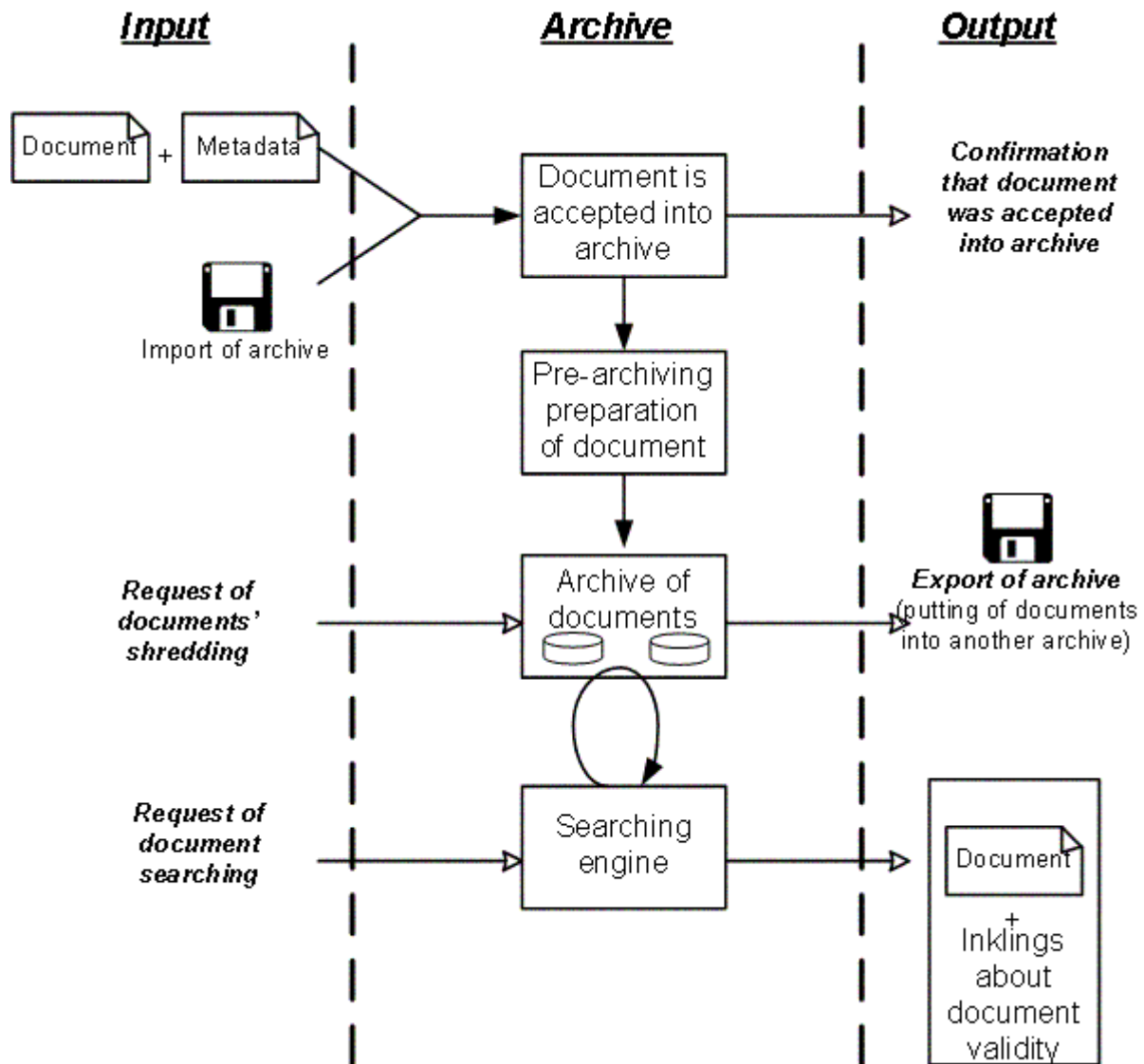
- Document name
- Keywords for searching for the document
- Archiving length
- Shredding type (give/not to give the document into the historical archive, automatic shredding or must be reviewed by an archivist before)
- Time length of renewals

5 Main functions of archive

If we do not think only about an archive theory but want to build the real archive, we may not solve only partial functions which the archive should have, but we need to build the complex archiving infrastructure which contains many partial services. These partial services must cooperate (interlock) to create the operational complex (unit).

If we look at the archive from larger distance (from the Moon), we can see the basic structure of an archive – picture 4. The main functions from the user's point of view are:

- **Storing of the document** into the archive and receiving of a trustworthy confirmation about the storage of the document into the archive. The archive can be entered either by source documents or even parts of lower-level archives.
- **Shredding of documents.** The part of shredding is to put chosen documents into the upper-level archive.
- **Searching for the document** in the archive. The result of this should be not only the particular document but also its inklings which prove the document validity. These inklings arose during document's renewals. Just as the part of the paper document there are some additional handwritten notes, seals etc. which create the "local color" of the document, the "local color" of the digital document is formed by various time stamps.
- **Canceling of an archive** – can happen in case of person's death, company bankruptcy or the canceling of an institution. Canceling of an archive causes the shredding of the whole archive i.e. export of all non-shredded documents including their metadata into another archive. The canceling of the whole archive is a special case of shredding when the whole archive is shredded.



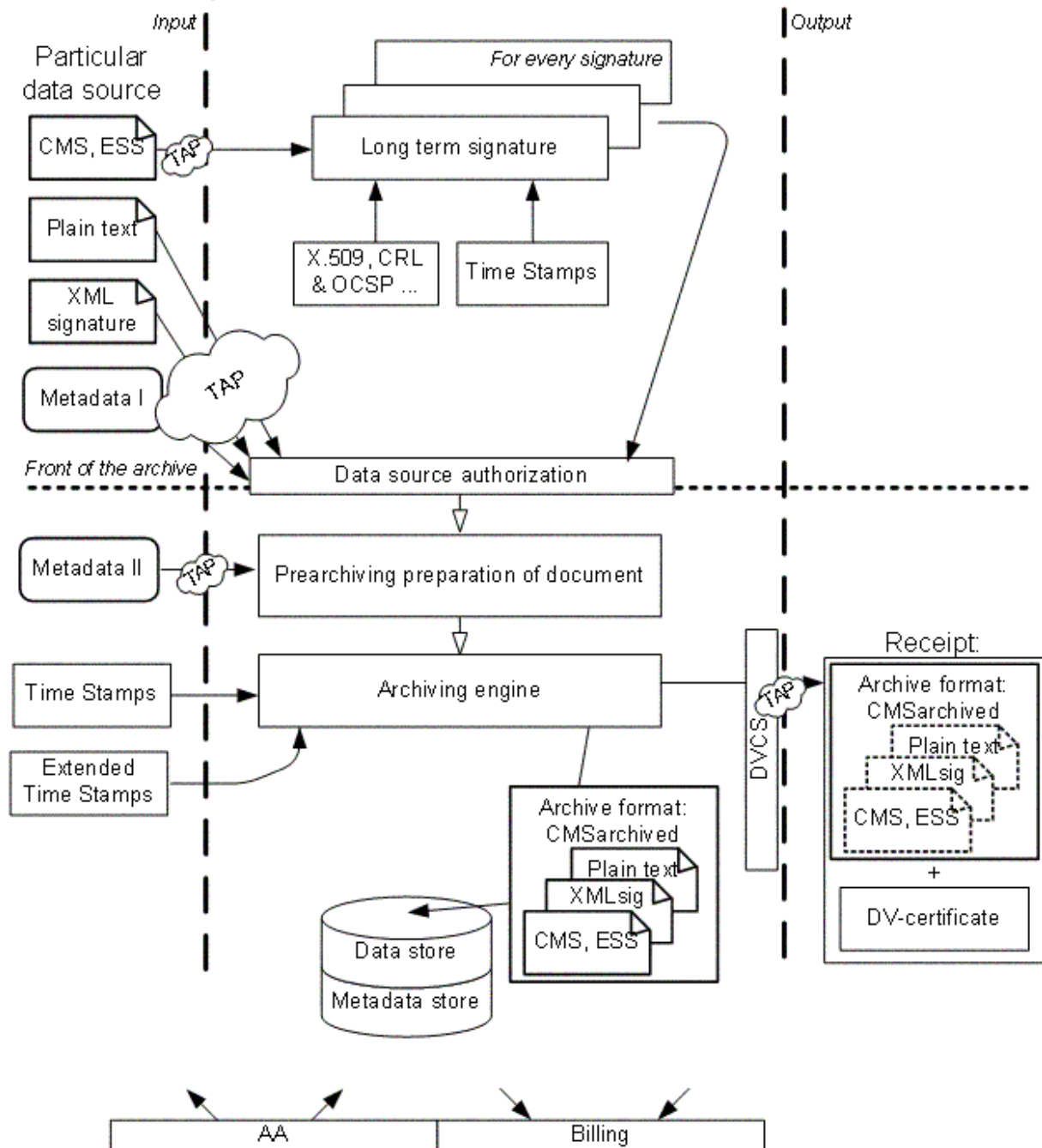
Picture 4 Main functions of archive from user's point of view

5.1 Confirmation

Archive must issue the following confirmations:

- Document was given into archive (i.e. archive accepted the document). And optionally that the archive archived the document (Registration number, stock number etc. Were given to it).
- The given copy of the document is owned by the concrete archive and the copy of the document is identical with the corresponding document which is stored in the archive.

5.2 Storing of document into archive



Picture 5 Storing of digital documents into archive

The following digital documents can enter the archive:

- Digitally signed documents. There are several standards of digital signature. Only format CMS/ESS is suitable for archiving. If document has another format of digital signature it seems to be a document which signature is not possible to renew. We must add the time stamp to such a document and renew the time stamp.
- Documents without digital signature but with the time stamp. Because the time stamp is digitally signed data structure, we can consider these documents to be digitally signed documents.
- Documents completely without digital signature. We should think over if to add the time stamp to these documents when they are accepted to the archive (i.e. something like a digital seal of the archive).

In case of digitally signed documents, the digital signature is appended acc. to [RFC-3126] (Electronic Signature Formats for long term electronic signatures) before archiving.

We must place an authorization module in front of an archive. It ensures the proof of identity of the data source. Attribute certificates seems to a suitable tool for this purpose.

Two metadata types enter the archive:

- Metadata given by the data source (metadata from the data source archive) containing e.g. the identification of the document within the data source archives.
- Metadata entered by the employees of the archive.

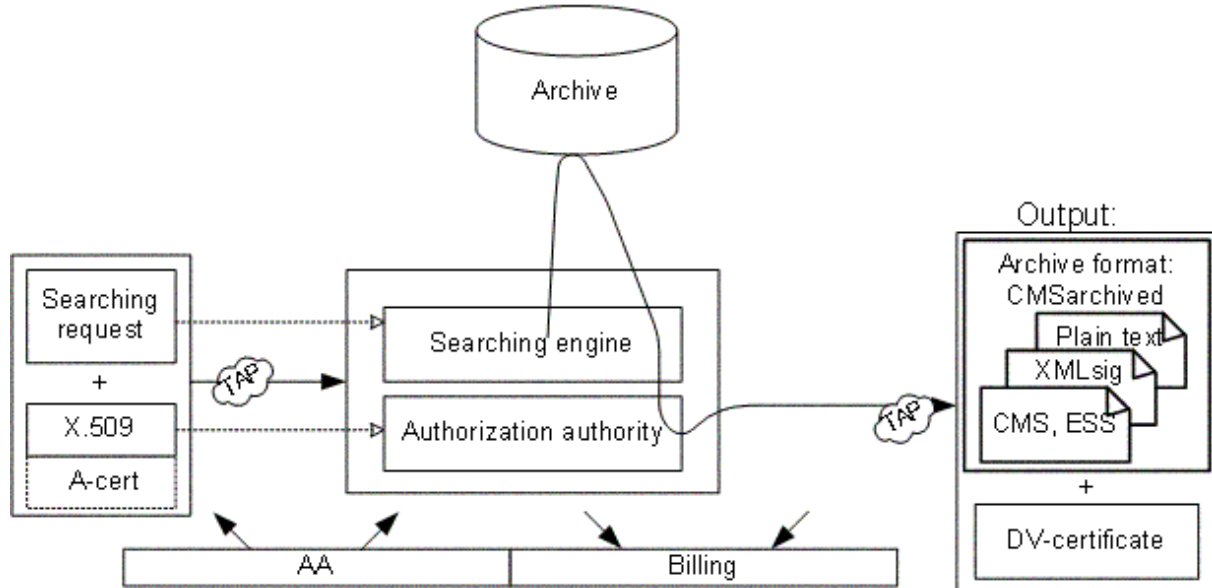
The DV certificate (cpd) seems to be the most suitable confirmation about accepting of the data into the archive.

Within the archive the document must be renewed with the help of time stamps and enhanced time stamps.

If modified in an appropriate way the TAP protocol [TAP] seems to be suitable access protocol for archive users. The format of archived document is also discussed in the Chapter 6.

5.3 Searching for document in archive

The client asks for searching for the document through the modified TAP protocol [TAP]. The authorization authority with the help of attribute certificates does the authorization.



Picture 6 Searching for documents in archive

The document with the DV-certificate, as a confirmation that the document was owned by the archive at a given time, would be an output. The document would also contain the inklings about its validity.

5.4 Shredding of document

Shredding of documents is not only the canceling of the document in the concrete archive, but in case of chosen document it represents also the export of the document including metadata into the upper-level archive.

The shredding reports are created by the archive and are stored in the digital form.

5.5 Renewal of document

Document renewal is one of internal functions of the archive. It is not visible from the user's point of view. User can see the renewal function of the archive only as additional inklings about the document validity.

6 Standard requirements

There are many protocols in pictures 5 and 6. Now we need to summarize the protocols requirements in the way to be able to build the real archive. We need to be involved in the following protocols:

- The format of the archived document from two points of view:
 - o The data format the document was created from (plain text, TIFF, XML etc.).
 - o The document format as a message stored in the archive (CMS).
- Types and formats of inklings about the document validity.
- The format of the confirmation which archive issues.
- The authentication of archive users.
- The protocol of the communication with the archive. It will support storing, searching, shredding, import/export and management of digital documents – including the management of the whole archive.

6.1 Data format

The main parameter important for the data format choice is the time how long it is necessary to take care for the document. I intentionally did not use “the time how long it is necessary to archive the document” to avoid misunderstandings. The document can be stored in one archive for some time and shredded after that. The shredding may not mean the document destruction but it can be only its export to the next archive. The document format must be chosen dependent on the time the document is in all archives.

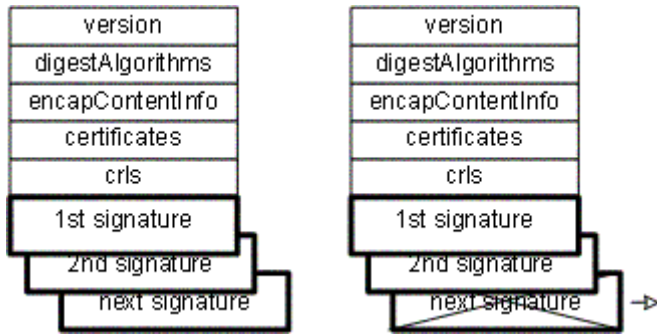
From this point of view we can distinguish two time lengths of the document storing:

- Document is stored only for such a short time, that neither emulation nor migration is required.
- Document will be stored so long it is probable that the programs needy for its viewing will not at disposal. It will be necessary either to migrate the document into the format readable for new programs or let the document as it is and emulate the old programs which are able to view it into the new environment. It means that the time for taking care of the document is long. Such documents we call the documents for the long-term archiving. The documents for the persistent archiving (i.e. documents of persistent historical value) are the special case of it.

The interesting exceptions are plain text or TIFF format for pictures. Their format is so simple that it is not necessary either to emulate or to migrate them.

I think that in case of long term or permanent value document it is necessary to use the plain text or TIFF format. These documents must be stored this way until other formats, which are not necessary to migrate, are defined. If formats plain text and TIFF are not sufficient in the future, I am sure that manipulation languages for description of digital texts for the long-term archiving will be created. These future manipulation languages will have to have one basic feature – the text will have to be readable even for laic. That is why I cannot recommend the XML or PDF formats which are used to view the text on a screen on to print it in a printer.

6.2 Message format



We suggest the format of a message, containing the digital document, which contains digital signatures as inklings about document validity in the CMS format.

Picture 7 CMS attack

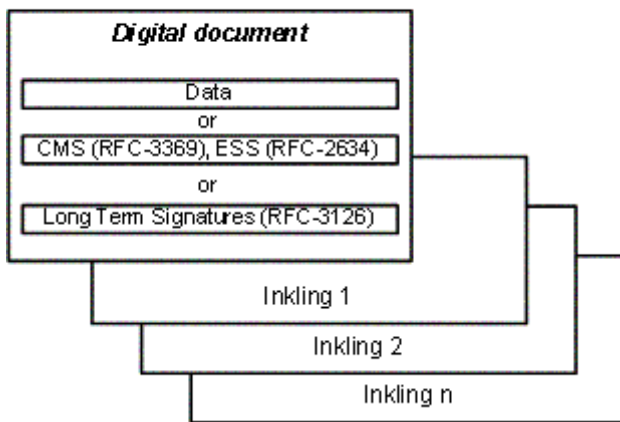
The main fault of these attempts is their dubious usage in case of documents which contain more parallel digital signatures. Of course it is possible to timestamp each of digital signature independently, but this method does not prevent the following potential attack: a hacker removes some of digital signatures including all their relevant time stamps.

The present attempts (e.g. [RFC-3126]) to include the inklings about the document validity into the digitally signed message append them in the form of non-signed attributes into the digital signature which can be removed by a hacker.

The main reason for them was that authors probably wanted not to create new data format to be able to process the data with the non-adjusted software. In case of more digital signatures all of them must be renewed independently (one by one).

The present attempts (e.g. [RFC-3126]) to include the inklings about the document validity into the digitally signed message append them in the form of non-signed attributes into the digital signature which can be removed by a hacker.

ArchivedData



In any case, special software is needed to process inklings. That is why we suggest sealing the digital documents into one CMS ArchivedData message in that way not to be possible to attack a message by e.g. erasing of one of its digital signatures..

Picture 8 ArchivedData type

The second reason of [RFC-3126] is probably that they want to prolong the validity of particular digital signatures. We think that it makes no sense. After we add all the data for verification of digital signature (what all relevant certificates are valid) into the document, it is no need to take care for particular signatures more. It is enough to add inklings about the existence of the document as a whole at a given time.

We suggest establishing new type ArchivedData which would be an analogy to SignedData type. The documents for which their inklings are stored would be encapsulated into this type. The inner part (eContent) would usually contain the SignedData or Data type.

The goal is to seal the original digitally signed or unsigned message into one unit which contains also inklings about validity of the document together with important metadata.

If an unsigned document (i.e. without digital signatures) should be archived, we usually will have to amend it with time stamp when archiving it. The time stamp becomes inkling about the time when the document was archived. This is a special example where the inner part of ArchivedData type is the Data type.

If an unsigned document (i.e. without digital signatures) should be archived, we usually will have to amend it with time stamp when archiving it. The time stamp becomes inkling about the time when the document was archived. This is a special example where the inner part of ArchivedData type is the Data type.

At first, we supposed that two types: `SealedData` and `ArchivedData` should be established. The `SealedData` type would have been useful when the document had been unsigned but we have needed to seal it with the time stamp or the DV-certificate.

Many documents are created unsigned or are created by people or systems that are not entitled to sign officially. But it is necessary to keep the proof that the document existed or better was owned by the concrete institution at a given time. The type similar to digital signature but having time stamp or similar inking instead, would fully fulfil these requirements.

The practical example of the situation when such a type is useful: an institution receives some document – it do not agree with the content but it needed to add some inking to the document confirming that the document exists at a given time. In this case the digital signature is not acceptable for the institution while the time stamp does not matter.

But when we put the `SealedData` and `ArchivedData` types in ASN.1 notation down on a paper, it was obvious that the `SealedData` type is only a subset of the `ArchivedData` type.

6.2.1 Archived-data Content Type

The archived-data content type consists of a content of any type or more document inkings. Any number of inkings in parallel can be added to any type of content. If we have more inking types bound to one document, they can be kept parallel.

The typical application of the archived-data content type represents one inking on content of the data content type. If the document is shredded and put into another archive, the next archive usually creates its own new inking(s).

The following object identifier identifies the signed-data content type:

```
id-archivedData OBJECT IDENTIFIER ::= { iso(1) member-body(2)
    us(840) rsadsi(113549) pkcs(1) pkcs7(7) x }
```

The archived-data content type shall have ASN.1 type `ArchivedData`:

```
ArchivedData ::= SEQUENCE {
    version CMSVersion,
    encapContentInfo EncapsulatedContentInfo,
    certificates [0] IMPLICIT CertificateSet OPTIONAL,
    crls [1] IMPLICIT CertificateRevocationLists OPTIONAL,
    inkingInfos InkingInfos }
```

```
InkingInfos ::= SET InkingInfo
```

The fields of the `ArchivedData` type have the following meanings:

`version` is the syntax version number. Current value is 1.

`encapContentInfo` contain the digital document (content). The content is represented in the type `EncapsulatedContentInfo`:

```
EncapsulatedContentInfo ::= SEQUENCE {
    eContentType ContentType,
    eContent [0] EXPLICIT OCTET STRING OPTIONAL }
```

```
ContentType ::= OBJECT IDENTIFIER
```

The fields of EncapsulatedContentInfo type have the same meanings as in case of SignedData type.

certificates is a collection of certificates.

crls is a collection of certificate revocation lists (CRLs).

Per-inkling information is represented in the type InklingInfo:

```
InklingInfo ::= SEQUENCE {
    version CMSVersion,
    inklingType INTEGER {
        timeStampToken          (0),
        -- RFC-3161
        enhancedTimeStamp       (1),
        -- based on linking hash
        dvc                      (2),
        -- RFC-3029
        ArchiveTimeStampsElement (3)
        -- Archive Time-Stamps Element
    }
    inklingAttrs [0] IMPLICIT InklingAttributes,
    inkling Inkling,
    uninklingAttrs [1] IMPLICIT UninklingAttributes OPTIONAL }

InklingAttributes ::= SET SIZE (1..MAX) OF Attribute
UninklingAttributes ::= SET SIZE (1..MAX) OF Attribute

Attribute ::= SEQUENCE {
    attrType OBJECT IDENTIFIER,
    attrValues SET OF AttributeValue }

AttributeValue ::= ANY

SignatureValue ::= OCTET STRING
```

version is the syntax version number. Current value is 1.

inklingType identifies the type of Inklinkg. The basic types are: time stamp, enhanced time stamp and Archive Time-Stamps Element.

inklingAttrs is a collection of attributes that are saved. The hash for Initial Time-Stamp for particular inkling is calculated from these attributes.

inklingAttributes MUST be DER encoded, even if the rest of the structure is BER encoded. Useful attribute types, such as signing are defined in next paragraph. The field MUST contain, at a minimum, the following two attributes:

- inkling is the appropriate inkling selected by inklingType. Inkling is calculated from the DER coded set of InklingAttributes.
- uninklingAttrs is a collection of attributes that are not included into inkling. The field is optional. This field is important for renewal of inkling.

The fields of `SignedAttribute` and `UnsignedAttribute` types have the following meanings:

- `attrType` indicates the type of attribute. It is an object identifier.
- `attrValues` is a set of values that comprise the attribute. The type of each value in the set can be determined uniquely by `attrType`. The `attrType` can impose restrictions on the number of items in the set.

6.2.2 Useful attributes

We can use some attributes established for the `SignedData` type:

- o The `content-type` attribute type specifies the content type of the `ContentInfo`. The `SignedData` and `Data Content` types will be mostly used.
- o The `message-digest` attribute type specifies the message digest of the `encapContentInfo` `eContent` OCTET STRING being archived. The attribute is intended mostly for an encapsulation of `Data` type messages.
- o The `signature time-stamp` attribute from [RFC-3126]. *It would be also useful to establish the attribute similar to this one which contains DVC instead of the time stamp.*

Next to it, it is necessary to establish new attributes:

6.2.2.1 Signer-Info

`Signer-Info` attribute contains one digital signature of an encapsulated message. The attribute can be used for an encapsulated `SignedData` type. This attribute can be here more times – one time for any particular digital signature of an encapsulated message - i.e. for each presence of the `Signer-Info` item in the encapsulated message.

The `Signer-Info` attribute MUST be an inkling attribute; it MUST NOT be an uninkling attribute.

The following object identifier identifies the message-digest attribute:

```
id-signer-Info OBJECT IDENTIFIER ::= { iso(1) member-body(2)
    us(840) rsadsi(113549) pkcs(1) pkcs9(9) x }
```

`Signer-Info` attribute values have ASN.1 type `SignerInfo` (as it is defined in the `SignedData` type):

```
Signer-Info ::= SignerInfo.
```

6.2.2.2 Metadata

This attribute contains important information about the document. It should be as safe and protected as the document alone. It can contain e.g. important information of a historian or a notary.

The following object identifier identifies the metadata attribute:

```
id-signer-Info OBJECT IDENTIFIER ::= { iso(1) member-body(2)
    us(840) rsadsi(113549) pkcs(1) pkcs9(9) x }
```

Metadata attribute values have ASN.1 type OCTET STRING:

```
Metadata ::= OCTET STRING.
```

6.3 *Type of Inklings*

The main inklings are:

- Time stamp acc. to [RFC-3161].
- DV- certificate acc. to [RFC-3029].
- Enhanced time stamp
- Archive Time-Stamps Syntax [ATS].

The first three types of inklings (Time stamp, DVC and enhanced time stamp) have sense in case of unsigned documents for which we want to create one-time inkling. These inklings are used to register the document at the time of its archiving. We do not suppose this inkling will be ever renewed.

In the opposite the ATS we will use in case of documents where we plan to renew their inklings.

6.4 *Archive Time-Stamps Syntax [ATS]*

It is an Internet draft this time. The goal of this suggested standard is mentioned in the Introduction of this standard:

„It is necessary to standardize data formats and processing procedures for such Archive Time-Stamps in order to be able to verify and communicate data preserving evidence. A first approach was made by IETF within [RFC-3126], where an optional Archive Time- Stamp Attribute was specified for integration in signatures according to the Cryptographic Messages Syntax (CMS) [RFC-3369].

Archive Time-Stamps Syntax (ATS) broadens and generalizes this approach for data of any format and takes requirements of practical use into account, especially handling a huge amount of data objects. ATS specifies syntax for Archive Time-Stamps Element, which contains Enhanced Time- Stamps and some additional data.“

The following terms are established:

„An Archive Time-Stamp is an Enhanced Time-Stamp used for long-term non-repudiation of data.“

```
EnhancedTimeStamp ::= SEQUENCE {
    digestAlgorithm AlgorithmIdentifier OPTIONAL,
    reducedHashtree [0] SEQUENCE OF {SEQUENCE OF OCTET STRING}
                                OPTIONAL,
    timeStamp          ContentInfo}
```

„An Archive Time-Stamp Chain is a timely ordered sequence of Enhanced Time-Stamps, where each Enhanced Time-Stamp preserves nonrepudiation of the Enhanced Time-Stamp before, although if it got invalid and until itself gets invalid. The process of generating such an Archive Time-Stamp Chain is called Time-Stamp Renewal.“

```
ArchiveTimeStampChain ::= SEQUENCE OF EnhancedTimeStamp
```

„An Archive Time-Stamp Sequence is a sequence of Archive Time-Stamp Chains, where each Archive Time-Stamp Chain preserves nonrepudiation of Archive Time-Stamp Chains before, although the hash algorithm used within Enhanced Time-Stamps hash-tree got weak. Nonrepudiation is preserved until the last Enhanced Time-Stamp of the last chain gets invalid. The process of generating such an Archive Time-Stamp Sequence is called Hash-Tree Renewal.“

```
ArchiveTimeStampSequence ::= SEQUENCE OF ArchiveTimeStampChain
```

„Archive Time-Stamps Element are all Archive Time-Stamps and other data, which are to be used to prove the existence of a data object or a data object group.“

```
ArchiveTimeStampsElement ::= SEQUENCE {  
    version                INTEGER { v1(1) },  
    usefulinformation      [1] OCTET STRING OPTIONAL,  
    encryption             [2] EncryptionMethod OPTIONAL,  
    archiveTimeStampSequence ArchiveTimeStampSequence}
```

The ArchiveTimeStampsElement type is a required inkling then.

6.4.1 Objections to ATS

Similar to the [RFC-3126], which is exclusively focused to time stamps acc. to [RFC-3161], the ATS is exclusively focused to enhanced time stamps. It is specified here: „An Archive Time-Stamp is an Enhanced Time-Stamp“

We recommend adding the possibility of time stamps acc. to [RFC-3161] to the ATS. This would enable to keep two kinds of independent inklings (two interindependent ArchiveTimeStampsElement) for one digital document. The first one will be based on time stamps acc. to the [RFC-3161] and the second one on enhanced time stamps. It could be also useful to use the DVC of the cpd type (see [RFC-3029]) instead of time stamps. Only the future will show which one of the inklings will be more useful for researchers. And more – if some new inkling type (new stamp type) is specified, it will be possible to add it.

I.e. we suggest the following ASN.1 syntax of the Archive Time-Stamp:

```
ArchiveTimeStamp ::= CHOICE {  
    [0]    enhancedTimeStamp EnhancedTimeStamp,  
    [1]    timeStampToken    TimeStampToken  
}
```

Then, an Archive Time-Stamp Chain will be a timely ordered sequence of Archive Time-Stamps, where.

```
ArchiveTimeStampChain ::= SEQUENCE OF ArchiveTimeStamp
```

If the Archive Time-Stamp Chain was created with the help of enhancedTimeStamp, the following article is involved in it.

If the Archive Time-Stamp Chain is consists of time stamps acc. to the [RFC-3161], then the messageImprint of the first time stamp is calculated from DER coded attributes inklingAttrs of the ArchivedData message. The messageImprint of next time stamps in this sequence is calculated from DER coded inklingAttrs attributes chained together with the content of messageImprint of all preceding time stamps in this sequence.

6.5 Enhanced time stamps

We think that the specification of the enhanced time stamp in the ATS is not worked deeply enough.

The enhanced time stamp should fulfill some features similar to the time stamp acc. to the [RFC-3166]:

- The enhanced time stamp should be also issued by the third independent party – the Enhanced Time Stamping Authority. It should have the identification of its issuer.
- The enhanced time stamp should be digitally signed by its issuer.
- The enhanced time stamp should have its unique identification.
- The enhanced time stamp request should be specified.
- We should choose more suitable name instead of the enhanced time stamp (e.g. time seal). This name is misleading and it is also hardly acceptable for [RFC-3166] authors.

The hash for the Initial Archive Time-Stamp should be calculated from the DER coded item `InklingAttributes`

The specification of enhanced time stamps which fulfils the mentioned conditions is written in details e.g. in the [Linking hash].

We recommend establishing the proposal standard for enhanced time stamp derived from the mentioned-above document.

6.6 Receipts issued by archive

It is either the receipt (confirmation) that the archive accepted the document or the receipt (confirmation) that the copy of the document issued by the archive is identical (validity of the copy) with the corresponding document stored in the archive.

The proof of it is the DVC (see [RFC-3026]). In practice, it is the `ArchivedData` message with the DVC inkling (the archive may be the issuer of the DVC).

6.7 Authentication of client entering archive

The authentication will be based on attribute certificates according to the [RFC-3281].

6.8 TAP

The TAP protocol [TAP] takes care for the communication between an user and an archive. The question is: Who is the user? The answer to this question is a list of roles which the user can play:

- The first putting of the document into the archive. An end user, an employee of a registry office or a Document Management System (DMS) can put a document into the archive..
- Import of document from another archive.
- Shredding and export of the document to another archive.
- Processing of the document done by an archivist. It contains the work with archive libraries (establishing of the library, work with metadata of the library), work with the catalogue of libraries, entering and changing of metadata, ordering of the document, cataloguing of the document and finally creating of inkling about the document validity and storing of the document into the archive.
- Research activities: searching for the document and picking up the document from the archive.

The renewals of inklings should not be the part of the TAP protocol. The renewal of inklings should be done by the archive itself – and the archive is from the point of view of a user (i.e. from the TAP protocol point of view) a black box. It means, with the help of the TAP protocol the requirements are poured into this black box, the black box grows something and tells its answer again through the TAP protocol.

6.8.1 Present TAP protocol draft

It presumes that either requests or responses are usually entered into the CMS of the `SignedData` type. The importance of this digital signature is in the authentication of the particular message, i.e. the entitled person sent the request and the genuine archive created the response. It is probable that this digital signature has no influence to the message validity.

Three operations are supported:

- First putting of the document into the archive. A request is expressed with the help of the `ArchiveSubmissionReq` type and a response with the help of the `ArchiveSubOrDelResp`.

```
ArchiveSubmissionReq ::= SEQUENCE
{
  version          TAPVersion DEFAULT v1,
  submitterName   GeneralName,
  policy          OBJECT IDENTIFIER OPTIONAL,
  archiveControls [0] ArchiveControls OPTIONAL,
  archivedData    ArchivedData
}
```

- Searching for the document. A request is expressed with the help of the ArchiveRetrievalReq type and a response with the help of the ArchiveRetrievalResp.

```
ArchiveSubOrDelResp ::= SEQUENCE
{
  version          TAPVersion DEFAULT v1,
  status          ArchiveStatus,
  archiveToken     ArchiveToken OPTIONAL,
  archiveControls [0] ArchiveControls OPTIONAL
}
```

- Cancelling of the document (no discussion about shredding is there) with the help of the ArchiveDeletionReq type. The ArchiveSubOrDelResp type is also used for a response.

The very important thing is the mechanism of the document identification in the archive. A user puts the document into the archive with the help of ArchiveSubmissionReq message and receives the ArchiveSubOrDelResp answer which contains the Archive token.

„ Archive token: an archive token is an object generated by the TAA when data is submitted and accepted for archiving. The archive token is returned to the submitter and may be used to request retrieval or deletion of the archived data and associated cryptographic information. For purposes of future retrieval or deletion, applications may treat the archive token as an opaque blob. The archive token includes: submitter DN, timestamp token, TAA date and time upon submission and, optionally, tracking information“

Archive token is defined by:

```
ArchiveToken ::= ContentInfo
  -- content type: id-tap-archiveToken
  -- content: ArchiveTokenData

ArchiveTokenData ::= SEQUENCE
{
  submitterName  GeneralName,
  timestamp      TimeStampToken,
  curTime        GeneralizedTime,
  trackingInfo    TrackingInfos OPTIONAL
}
```

The conclusion is that within the archive the document is identified with its time stamp and with the time when the Archive token was created.

This system is probably effective in case of these archives which are a part of some Document Management System DMS. The document is sent to the archive and it returns the Archive token which the DMS stores. Consequently, the DMS can even erase the document because it is safely stored in the archive.

When searching for the document, it is searched in the DMS. The Archive token is found here and according to this Archive token the document can be found.

But imagine that somebody will want to find out this document this way after 300years when the relevant DMS will have been cancelled for many decades. It is absurd. The researcher will look for some document – he will not be exactly sure which one – and the access key for the document would be a document hash!

6.8.2 Objections to TAP protocol

6.8.2.1 User's authentication

The document can be authenticated by its digital signature. But the client (user) should have the relevant attribute certificate to be entitled to do the particular operation. The reason of it is that no the statutory representative of the organization (whose certificate can have such competencies) but the common clerk will do the archiving of the particular documents.

6.8.2.2 Putting document into archive

We must realize that the following data are needed to be able to put the document into the archive:

- The document source identification (It should be either in the subject of the user certificate or in the relevant attribute of the attribute certificate).
- The document identification (is created by the archive when processing the document)
- Date and time
- Document name
- Keywords for searching for the document
- Length of archiving and shredding type
- Length of renewals (it is a matter of archive)

I.e. the request for archiving should be appended of items containing the following information:

- Document name
- Keywords for searching for the document
- Length of archiving and shredding type
- Other metadata

We suggest enlarging the `ArchiveSubmissionReq` in the following way:

```
ArchiveSubmissionReq ::= SEQUENCE
{
  version          TAPVersion DEFAULT v1,
  submitterName   GeneralName,
  policy          OBJECT IDENTIFIER OPTIONAL,
  archiveControls [0] ArchiveControls OPTIONAL,
  dokumentName   [1] DocumentName OPTIONAL,
  keywords       [2] Keywords OPTIONAL,
  archivationPeriod [3] ArchivationPeriod,
  nextMetadata   [4] nextMetadata OPTIONAL,
  archivedData   ArchivedData
}
```

Where:

```
DokumentName ::= SEQUENCE {
  [0] name      UTF8String OPTIONAL,
  [1] uri       IA5String OPTIONAL
}
```

The `name` contains the document name and the `uri` contains the uri where other document specifications can be found.

```
Keywords ::= SEQUENCE SIZE (1..MAX) OF Keyword
Keyword  ::= UTF8String
```

It contains keywords for fast searching for the document.

The `ArchivationPeriod` specifies the time of archiving (but the final decision is always on the archivist who specifies it when processing the document).

```
ArchivationPeriod ::= SEQUENCE {
    period  GeneralisedTime,
    action  INTEGER [
        erase (0), automatic (1), manual (2)]
}
```

where the `period` contains date and time of the shredding.

The `action` contains the way of shredding:

- `erase` – when the shredding time is over erase the document from the archive.
- `automatic` – when the shredding time is over prepare the document for the export to another archive.
- `manual` – when the shredding time is over inform the archivist that he must decide whether the document will be erased or exported.

```
NextMetadata ::= UTF8String.
```

The `NextMetadata` contains important information about the document (e.g. The document is suspected not to be genuine).

The DV-certificate [RFC-3029] should be also the receipt confirming that the document was given from the document source into the archive. This means that the `ArchiveSubOrDelResp` should be enlarged:

```
ArchiveSubOrDelResp ::= SEQUENCE
{
    version          TAPVersion DEFAULT v1,
    status           ArchiveStatus,
    archiveToken     ArchiveToken          OPTIONAL,
    archiveControls  [0] ArchiveControls  OPTIONAL
    dvc              [1] DVC                OPTIONAL
}
```

In the requester item in this DV-certificate there should be the distinguished name of the archive.

6.8.2.3 Import of document

The format of the imported document should be similar to the one used when the document was archived first. The only difference is that the archive should support the `ArchivedData` type.

6.8.2.4 Document shredding

Documents, which should be put into another archive during shredding, should be exported in the form which can be imported into another archive. It is the `ArchivedData` format entered into the `ArchiveSubmissionReq` message with the following filled items:

- Document Name
- Keywords
- Archiving Period

6.8.2.5 Processing of document by archivist

The document is automatically accepted/imported by the archive. The archive automatically issues the confirmation (DV-certificate) about the acceptance of the document.

The document accepted this way is inserted into an input queue waiting for being processed by an archivist.

The following must be at the disposal to the archivist:

- An interface for working with archiving libraries:
 - Creating of archiving library
 - Editing of metadata of archiving library (introduction, history, Archiving characteristics, brief analysis of the content, specification of the ordering of the library etc.). It is not needed to digitally sign every particular note, the complete list of all text reviews (audits) is enough. In this list we should be able to read the distinguish name of the archivist who made the audit.
 - Creating of the catalogue of the archiving library.
- To work with the catalogue of the archiving library including cataloguing of particular documents in the catalogue of the archiving library.
- Interface for the verification of inklings of documents.
- Interface for working with metadata of the particular document.
- Interface for adding of various inklings to the document, document cataloguing including adding of the stock number of the document and storing it into the archive.
- Interface to identify documents for shredding.

All the mentioned-above actions can be realized through the common web server (HTTPS protocol). In fact, the documents are not digitally signed but digitally stamped. Stamps are issued by an independent third party – the authority for stamping. The decision what to stamp is up to the archivist. The archivist only authenticates himself against the server and gives orders e.g. “add relevant inklings to the document”.

6.8.2.6 Research work

Research work is automatically divided into two parts:

- Searching for the document. At first, the archivist goes through the catalogue of archiving libraries, next, the catalogue of the particular library where he will find the stock number of the document. Knowing the stock number of the document he can finally ask for the particular document in the archive. It is also possible to make all of these operations with the help of the HTTPS protocol.
- The archive returns the particular document. The document including all of relevant inklings should be in such form to be able to use it as a proof at a court. The DV-certificate cpd or ccpd or the ArchivedData message, which contains these certificates, is in the requested form (fulfill requirements).

7 Conclusion

To be able to create the trustworthy archive effectively it is necessary to:

- Create the proposal standard of the DVCSP protocol

- Amend the type `ArchivedData` to the CMS
- Create the standard for enhanced time stamps.
- Update the ATS and TAP.

8 Acknowledgements

Thanks to Peter Sylvester for sharing information from the OpenEvidence project.

9 References

- [RFC-2026] Bradner, S., "The Internet Standards Process – Revision 3", BCP 9, RFC-2026, October 1996.
- [RFC-2028] Bradner, S. and R. Hovey, "The Organizations Involved in the IETF Standards Process", BCP 11, RFC-2028, October 1996.
- [RFC-2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC-2119, March 1997.
- [RFC-3029] Adams, C., Sylvester, P., Zolotarev, M. and R. Zuccherato, "Internet X.509 Public Key Infrastructure Data Validation and Certification Server Protocols", RFC-3029, August 2001.
- [RFC-3039] Santesson S., Polk W., Barzin, P., Nystrom, M., "Internet X.509 Public Key Infrastructure Qualified Certificates Profile", RFC-3039, January 2001.
- [RFC-3126] Adams, C. Pinkas, D. Ross, J. Pope, N., "Electronic Signature Formats for long term electronic signatures", RFC-3126, September 2001.
- [RFC-3161] Adams, C., Cain, P., Pinkas, D. and R. Zuccherato, "Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)", RFC-3161, August 2001.
- [RFC-3275] Eastlake D. 3rd, Reagle J., Solo D., "(Extensible Markup Language) XML-Signature Syntax and Processing", RFC-3275, March 2002.
- [RFC-3281] Farrell S., Housley R., "An Internet Attribute Certificate Profile for Authorization", April 2002.
- [RFC-3369] Housley, R., "Cryptographic Message Syntax (CMS)", RFC-3369, August 2002.
- [ATS] Brandner, R., Gondrom, T., Pordesch, U., Tielemann, M., "Archive Time-Stamp Syntax", Internet Draft <draft-brandner-et-al-ats-00.txt>, July 2003
- [TAP] Wallace, C., Chokhani, S., "Trusted Archive Protocol (TAP)", Internet Draft <draft-ietf-pkix-tap-00.txt>, February 2003
- [Linking hash] Cybernetica AS Tallinn Estonia, "Protocols and data formats for time-stamping service", http://www.timestamp.cyber.ee/timestamp_en.pdf, 2002

10 Authors' Addresses

Libor Dostalek
PVT, a.s.
Kovanecka 30

Marta Vohnoutova
PVT, a.s.
Kovanecka 30

190 00 Prague, Czech Republic
E-Mail: Libor.Dostalek@pvt.cz

190 00 Prague, Czech Republic
E-Mail: marta@nexta.cz