UNIVERSIDAD DE LA RIOJA

TESIS DOCTORAL

Título
Security in Petri nets sharing and storage: subnets, privacy, integrity, authentication and non repudiation
Autor/es
Iñigo León Samaniego
Director/es
Emilio Jiménez Macías y Juan Ignacio Latorre Biel
Facultad
Escuela Técnica Superior de Ingeniería Industrial
Titulación
Departamento
Ingeniería Eléctrica
Curso Académico
2014-2015

University of La Rioja

Doctoral Thesis

Security in Petri nets sharing and storage: subnets, privacy, integrity, authentication and non repudiation

Author:
Iñigo León

Advisors:
Dr. Emilio Jiménez
Dr. Juan Ignacio Latorre

A thesis submitted in fulfilment of the requirements for the degree of Doctor on Electrical Engineering, Mathematics and Computer Science

June 2015

Declaration of Authorship

I, Iñigo León, declare that this thesis titled, 'Security in Petri nets sharing and storage: subnets, privacy, integrity, authentication and non repudiation' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: July 2015 _____

UNIVERSITY OF LA RIOJA

Abstract

Faculty of Science, Agrifood Studies and Computer Science
Department of Electrical Engineering and
Department of Mathematics and Computer Science

Security in Petri nets sharing and storage: subnets, privacy, integrity, authentication and non repudiation

by Iñigo León

In this thesis I approach the study of Petri nets from the point of view of the security. There several goals in this thesis. First of all, I will define a subnetting process by building a framework of definitions and notations to create subnets from the original Petri net. Then, the creation of a PNML extension that allows the representation of subnets. In this work only the structure of the network is processed. The study of markings and properties of nets with hidden pieces will we analyzed in further works.

One application of this subnetting and PNML representation is the possibility of hiding part of a Petri net, facing a possible distribution, maintaining the privacy of the critical, secret, or complex parts of the system. However this hidden information is not eliminated from the net, but encrypted inside. Other application explained is the possibility of digital signature of subnets, providing security services to the net and/or subnets.

My original contribution to knowledge are:

1. Comprehensive study of subnets, abstracting their internal structure from the exterior by using front-ends. A method to build these subnets from the complete Petri net is explained and analyzed matrixed.

2. PNML has no way to represent subnets, so I approach a possible PNML extension to do it.

3. Subnetting and PNML extension to represents subnets allow to apply several security technics that offers encryption, data integrity, authentication and non repudiation

Acknowledgements

Thanks to my years of personal study and my curiosity for new knowledge, I fought to combine my job, my family and studies. I has been a difficult work, but the result has been worth it.

Thanks to my wife Nuria, and my daughters Valvanera and Blanca for putting up with me in my days of little sleep.

Thanks to my thesis advisors because without their help I couldn't have built this project.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background of the research

Petri nets are widespread for modeling many classes of systems, such as manufacturing logistics processes and services [1, 2], concurrent systems [3], etc. However, all these nets are described in a comprehensive way and must have the information of the entire net to determine its evolution. Furthermore, these nets can be modified with no control of integrity or authoring, for example.

1.2 Research problem

The problem occurs when somebody doesn't want to describe the whole subnet. Or, maybe, is wanted one part of the process to be only accessible for one specific person or entity.

The first approach to solve this problem is to take two Petri nets:

- one Petri Net with only the public information, extracting the private data. This is an incomplete model of the process
- another Petri Net with the whole information for the interested person or entity.

As you can notice, this is not an efficient way to publish this kind of Petri Nets.

Other problem appears when I want to protect parts of the net from undesired modifications or ensure the authoring of some parts (or the whole net).

1.3 Justification of the research

It would be interesting to provide security to a Petri net:

- hiding a part of it. This can be useful, for example, distributing a process we want to be secret [4], or simply to be a part of the net to be complex and do not interest handle for any reason [4].
- avoiding not allowed changes in it (or a part of it).
- authenticating it (or a part of it). Useful to ensure who has developed a Petri net or subnet.
- avoiding the possibility of supplant other people in the authority of the Petri net or some of its parts.

So here is my contribution. I have researched the possibilities of hiding a part of a Petri Net so that everybody can access the public information, maintaining the secret of the private data. This private data is accessible only for authorized people. And not only that: I ensure data integrity, authentication and non repudiation to Petri nets or subnets.

Some authors study the possibilities of Petri nets reduction [5, 6, 7, 8, 9, 10], grouping in one place or transition a subnet, so that what happens on this subnet, is encapsulated in a single point of execution. However, we want to go further by defining parts of the net that are hidden (not clustered) and what are the implications, studied within network properties.

The main objective of this thesis is to extract parts (subnets) of a Petri net and provide them of wide security (privacy, integrity, authentication and non repudiation).

1.4 Methodology

In order to achieve this goal, I have defined three milestones:

1. Extend Petri Nets in order to define subnets, abstracting the internal structure from the the rest of the net using front-ends, focussing on hiding information.
2. Choose a lossless and extendible representation of this kind of Petri Nets
3. Define a hiding and signing method for this representation

For the first milestone, I work for the creation of the theoretical basis for further study of Petri nets in which certain parts are hidden. So we setup a generic framework of definitions and notations that allow us to deepen in the study of the characteristics and properties of Petri nets and their subnets [11, 12]. Also mention work already carried by other researchers in which we rely for our goal (i.e. [3, 13, 14, 15]). All of this will be necessary to create the framework that allows us to study occultation in Petri nets. We will expand the vision of Petri nets, providing them with greater functionality, such us attachable subnet.

The next step in this work is to choose (or define) a flexible representation of Petri nets that allows us to translate the previous extended nets. This representation has to be really extendible and flexible in order to be able to show actual and future characteristics of Petri nets. I can advance you that the selected representation is the standard PNML and I have to define and extension for it in order to represent subnets that are going to be secured.

Once selected this representation, the last step is the hiding and signing method(digital signature provide integrity, authentication and no repudiation services). Once more, I bet for standard protocols like XMLEncryption and XMLSignature.

This is a very basic investigation because I extend the very early definitions of Petri nets. Because of it, the results of this thesis is very probably extensible to any other development whose base are the classic Petri Nets. For example, I am not going to study colored Petri nets, neither timed Petri nets, etc. But it is very easy to see that the results achieved in this thesis can be applied to them with little problems.

1.5 Delimitations of scope and key assumptions

For this work we will always deal with ordinary and pure networks, unless otherwise expressly. This assumption is only for clarity reasons, because the protocols and methods described in this work are perfectly extensible to other kind of Petri nets, as long as these Petri nets are representable in PNML format.

Chapter 2

Literature review

In this chapter I am going to go over the ancient Petri net history. This work is a very basic investigation on Petri nets. This means that the most of the references are quite general.

For this literature review, I will follow the structure of this thesis:

1. First of all, I am going to describe some generalities of Petri nets as an introduction.
2. Then I will describe subnets and the process of subnetting (splitting Petri nets into several subnets). Some of the defined subnets (or the complete net) are going to be secured.
3. The next step is to explain some possible Petri net representations and my choice of PNML for the securizing process.
4. Once selected PNML, I am going to extend this language to support the subnets defined before.
5. The last step is the securing properly speaking. To do this, I will use the standard XMLEncryption for ciphering the secret information and XMLSignature for integrity, authentication and non repudiation.

2.1 Introduction. Petri nets

In the 60's, Carl Adam Petri invented a new way to describe distributed systems called Petri nets [16, 17, 18]. Many net theories are based on those works[19]. Nowadays, Petri nets are really extended to represent discrete systems [20, 21, 22, 23, 24]. There are lots of applications of Petri nets:

- Modelling of sequential processes[25], concurrent systems[26, 27], manufacturing [13, 28, 29, 30, 31], logistic processes [2], discrete event systems [32], ...
- Simulation of industrial applications [33, 34], logistic and production systems [1],...

This work is not about Petri nets applications, so I am not going to deepen this field. However I really mind the intrinsic structure of them. Since the definition of Petri nets, many authors investigated about them. The basic basis of my work are some of the best Petri net researchers of the world, who are included in this review:

- Murata, T [11, 35, 36]
- Silva, M [12, 31, 37]
- Peterson, JL [15]

And not only general Petri nets. Any kind of extensions are well received

- Lien [38]: Generalized Petri nets
- Jensen, K [3, 26, 39]: High level Petri nets, coloured Petri nets
- Silva [24, 40]: Continuous Petri nets
- Khomenko, V [41]: High level Petri nets
- Ratzer [42]: Coloured Petri nets
- Kristensen [27, 43]: Coloured Petri nets
- Silva [44]: Fluid Petri nets
- Latorre [22, 23]: Aggregation Petri nets and coloured Petri nets
- Campos [45]: Stochastic Petri net models
- Recalde [46]: Continuous Petri nets
- David [14]: Discrete, continuous and hybrid Petri nets
- Fraca [47]: Fluid and untimed Petri nets
- Vazquez [48, 49]: Stochastic continuous Petri nets and fluid Petri nets

The study of subnetting and securing of all of them are outside the scope of this work, but as all of these Petri nets extensions are reproducible by a graph, they are susceptible of applying the same described methods with light variations (like color, arc types, ...).

Several authors studied properties of Petri nets and they have been very useful as [50, 51, 52, 53, 54]. But they don't contribute in the main goal of my work. They are general references for the theoretical development.

Other sources are authors that studied general theory on structure of systems, such as Teruel [55] is.

But this work is not oriented towards studying all of these Petri net type. I want my work so general that it is easily exported to practically any kind of existing Petri nets or future Petri nets types that maybe are not defined still.

2.2 Subnets

The study of subnets is very ancient with general works by Silva [12, 31], Murata [11, 35] and Peterson [15]. All of them have been very useful for my preparation and they are the theoretical basics of my work. But there are other more specific works that study several aspects of subnets.

The first approach was presented by Valette [5], who studied the possibility and properties of replacing places or transitions by subnets. Suzuki (in collaboration with Murata) [6] continued it with a method for expanding and reducing Petri nets. Basically, they wanted to substitute places and transitions by subnet and viceversa, maintaining the properties of the net. Druzhinin (and Yuditskii) [8] completed these works explaining how to construct regular Petri nets from standard subnets. But basically it is an algorithm to replace places or transitions with well-formed Petri nets. My contribution in this field is the description of a way to replace subnets (not only places or transitions) with other subnets. Their work can be seen as a particular case of my description of macroplace an macrotrasition. However I don't deepen the properties because it is not my goal. This topic can be object of further investigations.

Other important works in this section are Fahmy's ones [7, 9]. At first sight (only reading the title), it seems to be the same I explain in the chapter 3. But this is not true. These two articles by Famhy are about the analysis of large Petri nets by cutting them into pieces. The second one [9] is and extension of the first one [7]. The idea is "divide and conquer" over the net. The partitioning of a net is useful because it preserves the properties of the original whole net. He centers the interest on the characteristics of the

net and the partitions. But he doesn't make an extensive study of subnets, that is what I do in my work. Basically, it is something like the previous paragraph: it can be seen as a particular case of my study.

Other work of interest is Hsieh' one [56], where he analyzes non-ordinary Petri nets for flexible assembly/disassembly processes based on structural decomposition, but it is not a net decomposition but a process decomposition. So it has no more interest for my work.

The last entrance in this section is the work of Xia [10]. His objective is to encapsulate a subnet in one place or transition. Then he study several ways of simplify that subnet or erasing redundant information. These subnets are grouped into a single point of execution. I want to go further by defining hidden (not grouped) parts of the net. However, this work has much to do with macroplaces and macrotransitions introduced in chapter 3.

2.3 Petri net representation

Other important question in this thesis is the election of a Petri net representation that allows us to translate defined subnets into that format in order to apply posterior operations over those subnets.

Apart from the general works of Petri nets named before introduced by Petri, Murata, Peterson and Silva [11, 12, 15, 16, 17, 18, 31, 35, 36, 37], there are other specific articles in this field:

- Hura [57] introduced the state space representation of Petri nets, but with no possibility of subnet representation
- Anishimov and Perchuk [58] in their work "Representation of exchange protocols and Petri using finite sequential machine nets" didn't do exactly what the title says. They used High level Petri nets in order to define interaction protocols between two objects. These protocols are seen as more precise formal languages. Definitely, nothing seemed to my goal.
- Das [59] defined the reflexive incidence matrix (RIM) representation of Petri nets, The possibilities of represent subnets in this case is basically the same as normal incidence matrices. So my method can be applied in the same way with this representation.

- Malyugin [60] worked in an arithmetical representation of Petri nets, but, as I said with Hura, there is no easy implementation of subnet representation. So this representation is not useful for my work.
- Kaushal [61] introduced a new formulation for state equation representation for Petri nets. The problem is the same as with Hura and Malyugin: no subnet representation.
- Kiritsis and Xirouchakis [62] defined a new matrix implementation of Petri nets for process planning. As it is matrix representation my method can be applied in the same way, as I said with Das.

2.4 PNML

PNML is a W3C standard for representation of Petri nets. Because of that, the main source of information is internet. In particular the official site `www.pnml.org` [63] has all the necessary information to work with it.

After several years of study, in 2004 the ISO/IEC 15909-1 [64] appeared to define conceptually and mathematically a xml representation of Petri nets: PNML [65]. It is explained in a less formal way by Billington [66].

So all the Petri nets that I am able to draw can be stored in PNML. Because of that, it is one of the most supported formats in almost every program that draw Petri nets.

PNML [63] is an implementation defined by the standard ISO/IEC 15909-2:2014 [67] that complement the anterior ISO/IEC 15909-1. The goal of this standard ISO is to define a transfer format of Place/Transition nets, High-level Petri nets and Symmetric nets. However, it is designed in such way that it can be easily extended, so that other versions of Petri nets can be supported later.

2.5 PNML extensions

There are ways to define these extensions in a graphical way, using eclipse as intermediate named ePNK, explained by Kindler [68] and Hillah [69]. Additionally, Moutinho [70] and Ribeiro [71] defined several PNML extensions for several kinds of Petri nets.

But this is not the main goal of my work. My intention is to describe the subnet extension in a theoretical way. Once described, it could be implemented with that kind of tools.

2.6 Securize information on Petri nets

There is very little literature about this topic (except my own work [4]). There are a couple of articles that seems to threat the theme, but with a deeper study we can see that the achievements are not what I am looking for.

Mahulea [72] point the problem of observability of timed CPNs, but not with the same meaning of my vision. His problem is that in determined circumstances it is difficult to estimate the initial/ actual state/marking of a Petri net. Anything like my work.

By his side, Saabori [73] approach the topic of hiding but not in Petri net structure, but in states of the Petri net. So it doesn't help.

Other work that at first sight may be interesting is one article from Velilla [74] that define a mechanism for safe implementation of concurrent systems. But this safety is for the process itself, not for the safety of the implementation. So it doesn't contribute to my work.

My intention is to securize part of a Petri net in two ways: hide and sign. The way to achieve this goal is cutting the net in subnets, represent them in PNML and then one or more of this goals:

- hide it.
- avoid unwanted changes.
- ensure identity of the custodian of the Petri net.
- avoid the impersonation of the custodian.

But there is not literature about the PNML representation for subnets, so this is new knowledge.

2.7 Security: XMLEncryption and XMLSignature

Both XMLEncryption and XMLSignature are W3C recommendations.

XMLEncryption is a way to cipher xml content. And not only that. Actually, it is a way to encrypt any kind of information and store it in xml format. In the same way, XMLSignature is a standard mode to sign xml or non xml content.

There are thousands of articles and applications of it, but, as they are W3C recommendation, they are completely defined. The last version of XMLEncryption and XMLSignature definition are [75, 76] respectively.

In this work, I apply these technologies in order to achieve the next goals

- Privacy: hide a entire or a part of a petri net.
- Integrity: nobody should modify concrete parts of a Petri net without being detected.
- Authentication: guarantee that nobody can impersonate another, for example stealing the creation of a Petri net or part of it.
- Non repudiation: ensure that nobody but a concrete person has done something in a Petri net (or part of it), for example, sign a Petri net.

In particular, encryption provides privacy and digital signature provides integrity, authentication and non repudiation.

My contribution to knowledge in this case is to mix two different knowledge areas as Petri nets and information security (encryption/digital signature) are.

So there are no literature about this topic, because nobody has developed it until I do.

Chapter 3

Private information in Petri nets. Subnets

3.1 Introduction

By default, a Petri net is described in a comprehensive and public way. In this chapter I am going to describe a way to declare private information of a Petri net. This information is candidate to be hidden, complying in this way with the goal of privacy. Furthermore, this theory can be applied afterwards in order to sign parts of a Petri net (subnets) achieving the objectives of integrity, authentication and non repudiation.

First of all I have to create a like framework of definitions, properties and methodologies in order to achieve this goal. The main idea is the concept of subnet an its interaction front-end.

Once defined this subnets, the Petri net can be divided into public and private chunks only by ordering the places and transitions in one subnet or another.

3.2 Petri subnets

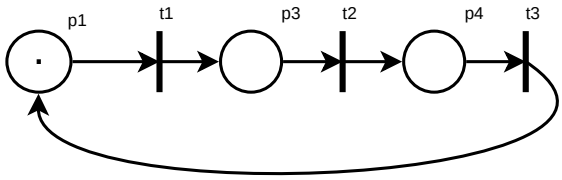
3.2.1 Definitions and properties

Let P and T the non-empty finite sets of places and transitions of a Petri net, respectively. Let $|P| = n$ (the number of places of the net) and $|T| = m$ (number of transitions of the net). Let α and β pre and post incidence matrices respectively. Let $N = \langle P, T, \alpha, \beta \rangle$ be a Petri net and let C the incidence matrix of N

we can notice that p_2 is an implicit place because its marking can be calculated as a function of p_3 y p_4:

$$M(p_2) = M(p_3) + M(p_4)$$

Moreover, by this same formula, it is clear that $M(p_2) \geq M(P_4)$ (marking cannot be negative) so the only place that can prevent enabling of T_3 is P_4. Thus eliminating p_2 does not alter the behavior of the network, which would be as follows:



In this network elements have been removed, no hidden. This example helps us to see the difference between hiding and a reduction.

3.4 Conclusions

As we have seen, any Petri net can be divided into any number of subnets, only limited by the number of places and transitions. Furthermore, each one of this Petri subnets has its own input and output front-ends in order to connect with the rest of the Petri net. Of course, the main application of this definitions in this thesis is that the private information of this Petri net is stored in one or more of these defined subnets.

So I have reached the first milestone: to extend Petri Nets definition in order to define the public and the private information.

Chapter 4

Petri net representation for subnets support. PNML

4.1 Introduction

Once explained how to split a net in several subnets, the next step is to define a way to secure one or more of those subnets.

There is not literature about this topic. Because of that I have to do a previous work about Petri net representation. First of all, a way to represent subnet must be defined. Depending on the selected representation, the way to occult subnets may be different or even impossible.

4.2 Petri net representations

There are four standard ways to represent Petri nets. Each one of them have their properties, advantages and disadvantages. But I want to select one that I am able to represent any kind of Petri net, its subnets and allow to hide information without erasing it.

4.2.1 Graphic representation

This is the clearest and extended way to represent Petri nets. It has a very important advantage and it is that a picture is worth a thousand words.

```
        </interface>
        <content id="N3_content">
          <place id="p6"/>
          <transition id="t4"/>
          <arc id="N3-a18" source="t4" target="p6"/>
        </content>
      </subnet>
      <place id="p1"/>
      <place id="p6"/>
      <transition id="t4"/>
      <transition id="t8"/>
      <arc id="a1" source="p1" target="N1-igp1"/>
      <arc id="a7" source="N1-ogt1" target="N3-igp1"/>
      <arc id="a11" source="N1-ogp1" target="p1"/>
      <arc id="a12" source="N3-ogt1" target="N2-igp1"/>
      <arc id="a16" source="N2-ogt1" target="t8"/>
      <arc id="a17" source="t8" target="p1"/>
    </page>
  </net>
</pnml>
```

Note that the interfaces are exactly equals than the b) net. N2 and N3 are the same, but N1 is different, but with the same interface (attachable net). So I could replace one subnet for another with no problem.

4.4 Conclusions

In this section we have seen a way to represent subnets in PNML format. It has not been a formal definition, but general guidelines in order to make a more extended study of the Petri nets. There are lots of different Petri net types. Each one of them has their own particularities that have to be translated into the PNML format. The method explained here is based on the basic general Petri nets, but it can see viewed as an algorithm that allow these types to define their own tags.

Basically, four new xml tags are introduced

- `<subnet>`, for delimitating the scope (places and transitions) of the subnet
- `<interface>`, for defining the gates to enter or leave the subnet. Each arc entering or leaving the subnet is associated to a specific gate. Each gate is represented with the tag `<gate>` and can be of several types: input/output (depending on if the arcs enters or leaves the subnet) and place/transition (depending on if the arc is associated to a place or a transition outside the subnet)

- `<content>`, where the places and transitions inside the subnet are placed, in addition to the arcs between them.

The really important thing in this chapter is to identify the subnet and process it in order to extract and interface that is the only way to enter and leave the subnet. The details of each one of the different Petri net types are not specified because of the big casuistry of them.

Once the method is applied, there has to be one subnet content and one subnet interface. There are two rules that must be accomplished:

1. no arc can join nodes inside the subnet with nodes outside and viceversa
2. the interface has to support all the arcs entering and leaving the subnet with the same information of the arcs replaced.

If these two rules are complied, then the subnet is correctly defined. Obviously, this method can be applied several times in order to declare several subnets. the only restriction is that each place or transition can be only in one subnet: no place/transition can be stored in two or more.

Chapter 5

Security

5.1 Introduction

This is the second main goal, after subnetting. Once the possible subnets are defined it is the turn of securing them. It is possible to secure Petri subnets or the entire net.

With secure, I mean four goals:

- **Privacy**. Concrete parts of the net must be occulted: the content is secret, so not everybody should be able to know it.
- **Integrity**. Any change in the secured parts has to be detected. If any of these parts suffers any kind of modification, the information may have been compromised, and perhaps it is not valid or correct. But I cannot know what has been modified: I can only detect that the original content has been changed.
- **Authentication**. I can authenticate the source of that net/subnet (the signer, author or guarantor).
- **Non repudiation**. With this characteristic, the possibility of supplant other people is avoided. So the person that sign that part cannot say that he hadn't done it. The signer cannot deny it.

There can be several reasons for hiding information of a Petri net. For example:

- One subnet is a secret process that I want to hide from indiscrete eyes
- I have a main process that communicates with other processes. These processes are susceptible to be changed and the only information I need is the interface, so they can be easily replaced by other implementations.

But this information should be accessible to authorized people without necessity of supplying any other kind of data. So the whole information may be stored in the same file.

In the same way, there can be many reasons for the rest of the security characteristics. For example, suppose that we have a Petri net that several people can access an modify and:

- Some parts of that Petri net have been validated an accepted, so I want nobody to change them. In this case **integrity** is needed.
- I want to know who has developed a concrete chunk of the net. **Authentication** is required.
- There is a part of the Petri net that is bad defined and go wrong. The person responsible of this part says that he hasn't made it and somebody has supplant him. Then, **non repudiation** is needed.

The best way to reach these goals is using standard and proved technologies. In this case, the selected technologies are:

- XMLEncryption[75] for privacy.
- XMLSignature [76] for integrity, authentication and non repudiation.

5.2 XMLEncryption

5.2.1 XMLEncryption revision

XMLEncryption is a World Wide Web Consortium (W3C) Recommendation for encrypting xml or non xml content. It is a standard xml file cipher. Both symmetric and asymmetric ciphering can be used, but in this case, symmetric is preferred. The main idea of this encryption is to replace the xml element or elements we want to be ciphered by other xml code that contains the ciphered data, in addition to information of the algorithms and keys used for the encryption process. When a non xml file is ciphered, the only option is to encrypt it completely. But, when it is applied to xml content, this technology allows us to define concrete fragments of the document we want to hide. Moreover, the xml document can be transformed before applying the encryption, for example, in order to format the normalize the xml content. In this work, the pieces of

xml content susceptible to be ciphered are, obviously, the subnets represented in PNML format.

Regardless of the data source (xml or non xml) the result is always a xml element. Normal is that this xml encrypted chunk have the whole necessary information to be decrypted. Among that information we can find:

- Ciphering algorithm: it is the name of chosen method to encrypt the data. It can be not included. In this case, both ciphering and deciphering agents have to know which is the exact this algorithm.
- The ciphered data: obviously this parts is mandatory and has always to be present.
- Name of the chosen key: it is optional. It is used when a set of keys is known by both ciphering and deciphering agents.
- Key: it is optional. In this case there is a symmetric key in order to encrypt the data and an additional pair of keys: one (known by the cipher agent) to encrypt the symmetric key and the other (known by the decipher agent) to decrypt it.

Actually, there are several options to apply XMLEncryption, such as the algorithm or the key. The exact election of those option values is responsibility of the Petri net sender. For example:

- Maybe both parts (sender and receiver) have a common set of keys, so they can use it in order to encrypt and decrypt the subnet content.
- Other common use is that the key is defined inside the options but it is ciphered itself. If the receiver have a pair of keys (public and private) and an asymmetrical algorithm (such as Diffie-Hellman or RSA) , the symmetric key can be ciphered by the sender with the receiver's public key. In this case, only the receiver can decrypt it using his private key.

This section does not want to be an extensive explanation about XMLEncryption but a general idea about its functionality. So I am not going to deepen the whole characteristics of XMLEncryption. The final decision about which options use is responsibility of those people that want to apply this work, basing their decision on the requirements of their own Petri net.

Once this is said, here we have a basic example of XMLEncryption. Let's take this original xml document:

```
<?xml version='1.0'?>
<PaymentInfo xmlns='http://example.org/paymentv2'>
  <Name>John Smith</Name>
  <CreditCard Limit='5,000' Currency='USD'>
    <Number>4019 2445 0277 5567</Number>
    <Issuer>Example Bank</Issuer>
    <Expiration>04/02</Expiration>
  </CreditCard>
</PaymentInfo>
```

LISTING 5.1: Clear xml content

In this example, we want to hide the credit card number <`Number`>. Several options are going to be applied: with and without the ciphering information

First of all let's see which will be the aspect of the ciphered content without information about ciphering, only replacing the clear data by the encrypted data: we have not information about the key or the ciphering algorithm. This is the xml ciphered code:

```
<?xml version='1.0'?>
<PaymentInfo xmlns='http://example.org/paymentv2'>
  <Name>John Smith</Name>
  <CreditCard Limit='5,000' Currency='USD'>
    <Number>
      <xenc:EncryptedData xmlns='http://www.w3.org/2001/04/xmlenc#'
          Type='http://www.w3.org/2001/04/xmlenc#Content'>
        <xenc:CipherData>
          <xenc:CipherValue>A23B45C56</CipherValue>
        </CipherData>
      </EncryptedData>
    </Number>
    <Issuer>Example Bank</Issuer>
    <Expiration>04/02</Expiration>
  </CreditCard>
</PaymentInfo>
```

LISTING 5.2: Ciphered xml content without ciphering information

As we can see, the credit card number has been replaced by a new tag <`EncryptedData`> that contains the ciphered credit card number.

Note. A new namespace `xenc` appear that is the standard namespace for XMLEncryption. However, in further examples this namespace can be delete for clarity and for space problems without loss of generality.

And now let's see how does this same example with information about the algorithm:

```
<?xml version='1.0'?>
<PaymentInfo xmlns='http://example.org/paymentv2'>
  <Name>John Smith</Name>
  <CreditCard Limit='5,000' Currency='USD'>
```

```
    <Number>
      <xenc:EncryptedData xmlns='http://www.w3.org/2001/04/xmlenc#'
          Type='http://www.w3.org/2001/04/xmlenc#Content'>
        <xenc:EncryptionMethod
            Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"
            xmlns:xenc="http://www.w3.org/2001/04/xmlenc#" />
        <xenc:CipherData>
          <xenc:CipherValue>A23B45C56</CipherValue>
        </CipherData>
      </EncryptedData>
    </Number>
    <Issuer>Example Bank</Issuer>
    <Expiration>04/02</Expiration>
  </CreditCard>
</PaymentInfo>
```

Listing 5.3: Ciphered xml content with algorithm information

As we can see, a new tag <`EncryptedMethod`> has appeared inside the <`EncryptedData`> tag with the algorithm used to cipher. In this case is `aes128-cbc`.

There is other method of Encryption that cipher the tag too. In this case, we would have the next code:

```
<?xml version='1.0'?>
<PaymentInfo xmlns='http://example.org/paymentv2'>
  <Name>John Smith</Name>
  <CreditCard Limit='5,000' Currency='USD'>
    <xenc:EncryptedData xmlns='http://www.w3.org/2001/04/xmlenc#'
        Type='http://www.w3.org/2001/04/xmlenc#Element'>
      <xenc:EncryptionMethod
          Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"
          xmlns:xenc="http://www.w3.org/2001/04/xmlenc#" />
      <xenc:CipherData>
        <xenc:CipherValue>A223B3B493G5C569M</CipherValue>
      </CipherData>
    </EncryptedData>
    <Issuer>Example Bank</Issuer>
    <Expiration>04/02</Expiration>
  </CreditCard>
</PaymentInfo>
```

Listing 5.4: Ciphered xml content including the tag itself

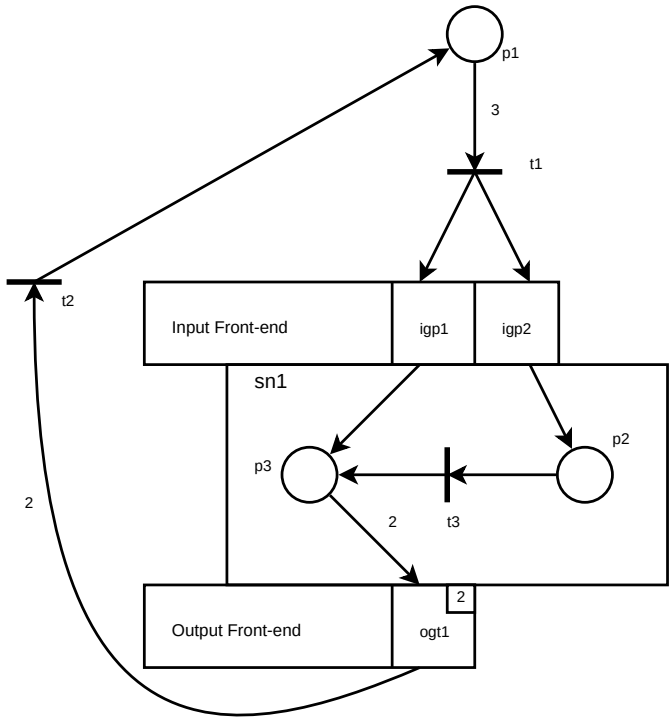
As we can see, the tag <`Number`> has disappeared and it has been included into the <`CipherValue`> of the <`CipherData`>.

This is a little approach to XMLEncryption functionality, but enough for understanding the next section.

5.2.2 XMLEncryption and Petri nets

Once described XMLEncryption it is time to apply it in order to hide part of a Petri net. Remembering the chapter 4, we have one Petri net with one or more subnets represented in a PNML file. These subnets are represented by a `<subnet>` tag that contains `<interface>` and `<content>`. This last tag contains the xml content that is going to be ciphered. Obviously, if we encrypt the interface we will have no way to connect the subnet with the rest of the net.

Let's take back the example used to explain the process of subnetting in the figure 4.6...



...and its PNML representation

```
<subnet id="sn1">
  <interface id="sn1-interface">
    <gate id="igp1" action="input" type="place"/>
    <gate id="igp2" action="input" type="place"/>
    <gate id="ogt1" action="output" type="transition">
      <inscription>
        <text> 2 </text>
      </inscription>
    </gate>
  </interface>
  <content id="sn1-content">
    <place id="p2"/>
    <place id="p3"/>
    <transition id="t3"/>
    <arc id="sn1-a2" source="igp2" target="p2"/>
```

```
    <arc id="sn1-a3" source="igp1" target="p3"/>
    <arc id="sn1-a4" source="p3" target="ogt1">
      <inscription>
        <text> 2 </text>
      </inscription>
    </arc>
    <arc id="a5" source="t3" target="p3"/>
    <arc id="a6" source="p2" target="t3"/>
  </content>
</subnet>
<place id="p1"/>
<transition id="t1"/>
<transition id="t2"/>
<arc id="a1" source="p1" target="t1">
  <inscription>
    <text> 3 </text>
  </inscription>
</arc>
<arc id="a2" source="t1" target="igp2"/>
<arc id="a3" source="t1" target="igp1"/>
<arc id="a4" source="ogt1" target="t2">
  <inscription>
    <text> 2 </text>
  </inscription>
</arc>
<arc id="a7" source="t2" target="p1"/>
```

The goal is to hide the internal content of the subnet. If we apply XMLEncryption to the data contained inside the tag <`content`>, we will get something like this, depending on the algorithm and key selected for the ciphering.

```
<subnet id="sn1">
  <interface id="sn1-interface">
    <gate id="igp1" action="input" type="place"/>
    <gate id="igp2" action="input" type="place"/>
    <gate id="ogt1" action="output" type="transition">
      <inscription>
        <text> 2 </text>
      </inscription>
    </gate>
  </interface>
  <content id="sn1-content">
    <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
        Type="http://www.w3.org/2001/04/xmlenc#Element">
        <xenc:EncryptionMethod
            Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"
            xmlns:xenc="http://www.w3.org/2001/04/xmlenc#" />
        <xenc:CipherData
            xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
            <xenc:CipherValue
                xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                Wr1njyJlYYOM9lAYqcwGCWkw2L4pUjQD2GGVoU9lVZ0wKqHY8y3lGY8FY4i5K
                3GY8FY4i5K3G8grIe1HRFqe7RtkFiXZgGMeYnQp6oB6ckKp3KFKHVqtucc9rA
                VzOgC7XAwe61HRFqe6RRVzXjNM9hlVZ0wKqHY8y3l3GY8FY4i5K3G8grIe2xN
```

```
                4u7x7fRtkFiXZgGMeYnQp6oB6ckKp3KFRRVzXjNAtVzOgC7XAw/oe61HRFqe6
                RRVzXjNMLU5ZgGMeYny8NVPQmUSDX7NRtnR6YnQp6oB6GY8F=
            </xenc:CipherValue>
        </xenc:CipherData>
    </xenc:EncryptedData>
  </content>
</subnet>
<place id="p1"/>
<transition id="t1"/>
<transition id="t2"/>
<arc id="a1" source="p1" target="t1">
  <inscription>
    <text> 3 </text>
  </inscription>
</arc>
<arc id="a2" source="t1" target="igp2"/>
<arc id="a3" source="t1" target="igp1"/>
<arc id="a4" source="ogt1" target="t2">
  <inscription>
    <text> 2 </text>
  </inscription>
</arc>
<arc id="a7" source="t2" target="p1"/>
```

If we try to represent this Petri net we will have the interface of the subnet, but the content is a black box as shown in the figure 5.1.

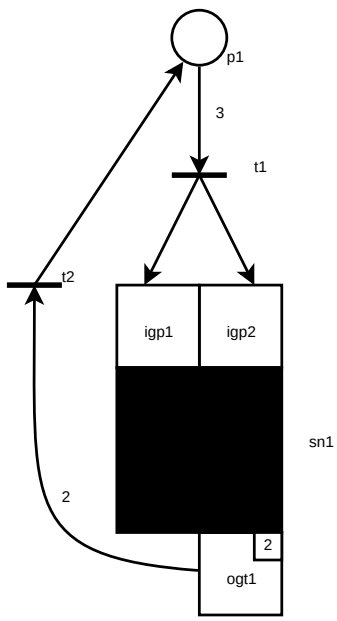


FIGURE 5.1: Petri net with hidden subnet

One important thing is that I can cipher several subnets of the same Petri net with distinct options. For example, if there are two subnets for both two distinct receivers each one of the subnets can be configured in order to each subnet can be decrypted by its own receiver.

5.2.3 Examples

5.2.3.1 Hiding several subnets

Let's take the Petri net from the figure 4.8. I want to occult the content of the subnets N1 and N2. The result should be, graphically, like this:

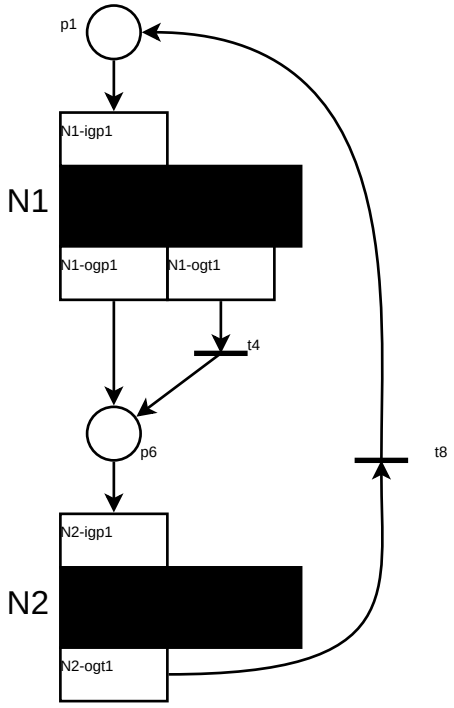


FIGURE 5.2: Petri net with two hidden subnets

The PNML content is this:

```
<?xml version="1.0" encoding="UTF-8"?>
<pnml xmlns="http://www.pnml.org/version-2009/grammar/pnml">
  <net id="latorre1" type="http://www.pnml.org/version-2009/grammar/ptnet">
    <page id="page1">
      <subnet id="N1">
        <interface id="N1-interface">
          <gate action="input" id="N1-igp1" type="place"/>
          <gate action="output" id="N1-ogt1" type="transition"/>
```

```
          <gate action="output" id="N1-ogp1" type="place"/>
        </interface>
        <content id="N1_content">
          <xenc:EncryptedData
              xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
              Type="http://www.w3.org/2001/04/xmlenc#Content">
            <xenc:EncryptionMethod
                Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <xenc:EncryptedKey xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                <xenc:EncryptionMethod
                    Algorithm="http://www.w3.org/2001/04/xmlenc#kw-tripledes"/>
                <xenc:CipherData>
                  <xenc:CipherValue>
                    2F3hsIebAicJ6WaS34Hy0OGJKFMAaOoTel/n4jfctbg=
                  </xenc:CipherValue>
                </xenc:CipherData>
              </xenc:EncryptedKey>
            </ds:KeyInfo>
            <xenc:CipherData>
              <xenc:CipherValue>
7oMx5W4VDF6fzvGcvRl71evbyDjTlSRtRVeiNEQSywmWMKz8tunVQPc3uATf4RcuGVroBFNt/3wn
PPSw6uXNPd9CSaTE0qGLmLlmWMBx3ge8rZlohS7uXGwq+Nfvc9QDmtl4+p5KpQdCyp1F/wBhVkGH
ezDWEsSH4fRoLcIwBmHcvroUCNyZ+6Un2+BLWr0U6xl0V9iyCuvZhUKjASupBZ/M3V7s6VzrzPtr
vvxjrIV3dcIZ1dAFBa3CxGjKMF76dqV9x1x3T9S7BqXLdbpXRYfcjOtbeDvIk3Y/HzJAQjGZaVfB
b6fye7aNiBlXOLLOx18W4OvJc9Y66Q3oAuVNlsSlrZAcxmshbDKD0FzkPE/QP929YOEerIq11KGw
SmJOhxsfZuSx+2KL4ZfrCx7CIpO1lJyCKu9r4PL7xZt9rVmKJcIAX546Mks3QKO94cBiQOCh8GbE
hbC7mzPMN/U24hIZ5KUMDdLibwsDchs5abXD+DweEMmC2AV328lLhott1fZRLb8lZ+rtrrmvnHIM
ZF+lrmjfhL9+1PAsVyMBsY8oN7gdXsxzxNHUAhrq0zFbIA58Ro4YjeRZhuACmXx/1y9wYgNjKSwo
x+w17hoYCUdwkzM12iiBJa/QcZcAALcLX2RTF9McAJOElonRrNdUgi9SCBz5ZblgONGyC7Edla7f
7P+QCFJaGFAoKYmDZF9OjTBQ4q+9FV+8/sUXSxqRnXUeUZEB7rhgVY68gyCp4L21aikXyEwQ2PkR
/GPZOWz/Yb7Sbq1pN/G6wqNbsepKWG9EV6n9rjSfiOocvy9wL8m1I0HmAp2l4FRTGKXLBluf0in+
7XvQIwaKz1dyoHGFESjr2Yt316K7LFakCqO6EC5dTIm1TwSuYuLOcNyz+HIOR1fPG1t1gbDok55R
WmJpZGtPciIWRjadmqyCSLHCgzMjAeIuUtf2GAowKsmTep7fUZ5jq3I59ggSKW9JCgjtc7oePQpG
lYxWfqO4oIlgONh+cIJKJMx+8VYh2a68GIkhn7816X2JlHNFjThlHF2rc31O17Wa6MA2eMO8zeyQ
ABsoUCx7BGHc0pzjxs5RF/lRh7rK7isFqUdDTgJAMX+bowZsGD2gDg93N/Yq/D/j0V4AcmWy5dAi
BmwY1WOPVjhgs0SIZhqfMvXo2o1pZvDRtvXEKALmVG6oBKZ+dQdLAIZW5xRlijYAEcwKfbTxosxE
1Gh5nw0tMAO9VxL9e1v1iW+ZdU9SLRf5FYULZ2+DtCqRQQvTP9AK5lJsh4rzV/f+YqQg1qq5IpMO
6OA3Z+OZcHtbZuRz1WyJM+240c8l9NpZILRYADj6Vz5/4FKmz7h1DGAr+TfBjqky7ZTqI+CzGQ==
              </xenc:CipherValue>
            </xenc:CipherData>
          </xenc:EncryptedData>
        </content>
      </subnet>
      <subnet id="N2">
        <interface id="N2-interface">
          <gate action="input" id="N2-igp1" type="place"/>
          <gate action="output" id="N2-ogt1" type="transition"/>
        </interface>
        <content id="N2_content">
          <xenc:EncryptedData
              xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
              Type="http://www.w3.org/2001/04/xmlenc#Content">
            <xenc:EncryptionMethod
                Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
```

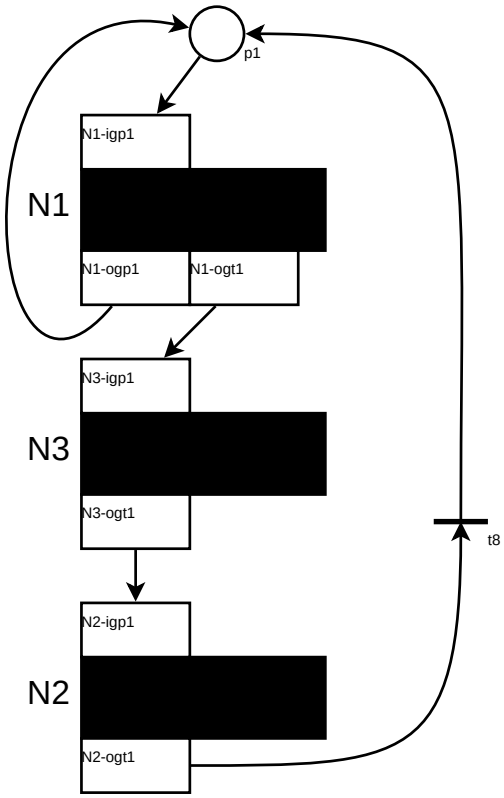
```
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <xenc:EncryptedKey
                  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                <xenc:EncryptionMethod
                    Algorithm="http://www.w3.org/2001/04/xmlenc#kw-tripledes"/>
                <xenc:CipherData>
                  <xenc:CipherValue>
                    2F3hsIebAicJ6WaS34Hy0OGJKFMAaOoTel/n4jfctbg=
                  </xenc:CipherValue>
                </xenc:CipherData>
              </xenc:EncryptedKey>
            </ds:KeyInfo>
            <xenc:CipherData>
              <xenc:CipherValue>
QbzkD9xHRIQVtHrpUQy4YHWcEuOXK135jCtnFHuHOTX4t9Bjwo29YR5GjmGNOjADZJvvQYcXasC3
PsBR/nymLyGvgnp8B/KNO6flN7f3FWGqPTo2oNpcckAryqCoi0Wy9wB6m/AfCAdVvmiUKzktEmLJ
Z8nyPk3n2pKyoKqLHSxtp1mu2Ll2gk6xpbdIgzy2aIT0KryrbRgcdLSSCI/L8om5d9MTstZGWVdr
2zl0ClAW68ef7/aqn6cyqJn/7czQvY8APFtQhWbQnBXjDMVCm1UH0FwfIffQfYENzi/8EhPB266P
ggj/2dv/Uk0TI+WB3BLhnbnrLoa0yIigSzYJYEz6EEn6D5deGOhnITxRdF9rgQliNxnirEeb+9ki
KlxFSB54h1TGzFV5UYQdHcuy0zf0XWBC2fQmdiCQtuzpvJqoOnQsdGmJhJjf65pZnFnvpNAPQF+S
RGN0tf4v8wP5ItSGl600GcSD2GRfkHfpfck6vAo/jxfzUs8j4qEOm9l9MB0TS1JLtQuNbBmTFTRz
3UgxChaDjBzR5s29uca9ZqirdZwnars+telyOVFD5qtBnWDTTVxSCLaeS7CE0gVWj9tQbUq7yqPK
fRnWhFyTZIuMCAuN+ybN3AWxsqwv499Q
              </xenc:CipherValue>
            </xenc:CipherData>
          </xenc:EncryptedData>
        </content>
      </subnet>
      <place id="p1"/>
      <place id="p6"/>
      <transition id="t4"/>
      <transition id="t8"/>
      <arc id="a1" source="p1" target="N1-igp1"/>
      <arc id="a7" source="N1-ogt1" target="t4"/>
      <arc id="a11" source="N1-ogp1" target="p6"/>
      <arc id="a12" source="p6" target="N2-igp1"/>
      <arc id="a16" source="N2-ogt1" target="t8"/>
      <arc id="a17" source="t8" target="p1"/>
      <arc id="a18" source="t4" target="p6"/>
    </page>
  </net>
</pnml>
```


We can see that the content of the subnets has been replaced by a new XML content with ciphered information that cannot be decrypted unless we have the correct decrypting key.

5.2.3.2 Encrypted replacement

Other application of this encryption method is that I can replace ciphered content with other ciphered content without decrypting it. For example, let's take the net 4.9 and

let's cipher it. As in the previous example, the graphical representation is:



And the PNML content is:

```
<?xml version="1.0" encoding="UTF-8"?>
<pnml xmlns="http://www.pnml.org/version-2009/grammar/pnml">
  <net id="latorre1" type="http://www.pnml.org/version-2009/grammar/ptnet">
    <page id="page1">
      <subnet id="N1">
        <interface id="N1-interface">
          <gate action="input" id="N1-igp1" type="place"/>
          <gate action="output" id="N1-ogt1" type="transition"/>
          <gate action="output" id="N1-ogp1" type="place"/>
        </interface>
        <content id="N1_content">
          <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
              Type="http://www.w3.org/2001/04/xmlenc#Content">
            <xenc:EncryptionMethod
                Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <xenc:EncryptedKey
                  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                <xenc:EncryptionMethod
                    Algorithm="http://www.w3.org/2001/04/xmlenc#kw-tripledes"/>
                <xenc:CipherData>
                  <xenc:CipherValue>
                    adSAmnteqXFUBdX3I6ClLoe27subXx/IPie9RUh3Viw=
                  </xenc:CipherValue>
                </xenc:CipherData>
```

```
              </xenc:EncryptedKey>
            </ds:KeyInfo>
            <xenc:CipherData>
              <xenc:CipherValue>
49adWIMQ3edEv8SArCsR6090GQU+2pjRhvkMH/kOlHDVcJmGw5Edu7jzIvZc7FF/4vzfKy8hgdXi
MIEQLhd7dbLdoBmkbZzZBnEZbD+JELC3lenV328xv0Y5jtx60vDM+cOEY94C8ctSlGmswlTC7m/h
pXyXtMtAipKcPBtqU6HiXW90gpEIosnEhat6+hyjpESgiQgbuHkYmSeCCqwMWvIR++UkB7Sf6hzP
er4QLUwZfuAbLH9NbReLmLbjpumGhWnMbvKXvwYAWnl1sjrtqnU2NZBWBLV8Uo3EMrTajI/2KA6o
XBo/piONJDEK5Q1lkpLm/+RFD+V1D7iKIXzUVdkOx4RbpOQarrz6bfnwfUBpPSDT+W1qz+8IILmd
DfXvyKGAefKzeOMiHf3YnSso+a5rm61XSmTVClBv2kYqcuADuh9GMsSVG1Uu94eg2C67DWKXFADK
dELk4g/W7FFDn77pqtcSGOBRb4NbGWa0z+m+hrXm/C8S8Ha/36A2DV1ysoAjrJlWsSxIPqf9gxL+
vWXpBr1d7KdD93g15natV4X80BD+2sWu23ZUlSavjLsOfNcocxIzlSXWlwGYyHoHGuHPcRzjn8Sr
hQ4s4gypOKlisc6EgRjbc+UlvLkVnc91LHHWKOhJuvCdgiodl6w//qf7J8X6ase0fhFBkHIUoOMz
XOQekOPKRgzHgz3iIkynwzShSpxSIaADynlJq6zUbbN77aOV4JyXJQRvyizZBCQz/kea8r1zBHJU
hPafRtTW1gZADB2C6xhrkE2KmMXlPzRwDGzPcRIfMo1frpg8PwJkNUhoFZSiZVvTOioB/jf4ITIO
Z/ffRxNokOVWHInjzjE09S5vyDli8V0q2fe1SOnjp/vKGZgv6cYxh6TrtjKGsC8J3nGgIb6PcSKR
+hzGCr8ZwGL3W8O5HogAELLDOC/u6knAdsOwCVPFcK72NroFPJu5O9ulGAJqNUoMlXLMAq1PfPmL
zdJPFcHSglOW/QvVtB3i7CyZrakyE2AiGZfmJko+pzFOI9VqkN989XHkchpF6gcQ1Zwcnxf8bben
fqT5zUVqStA4xJGCVHtyWIkIWGdbmIpzdEpiDUuozj+HzmC2zTS3r7EaViii34TOJkbOFt6IfTDn
NNX+LeH8P3JGxKEv27drQmchpWXrvithwt78csYPB6G395Wnpo+joQcQ7Bv7TREpu1n+0KdydZ3K
aGyozTHOkGMKoL1NCAVoj9f/4chjn/k9NM58KF63F0BtIDWS70k69iQb2rs5kAyA2SJ7YCkE6OI3
LXgAFjgOrXqjF9NSlwt3eEFsDWRdhn/epVGbkexdnjXBOnGW79AFcClYsVkj1vsSmJx5EvkvIw==
              </xenc:CipherValue>
            </xenc:CipherData>
          </xenc:EncryptedData>
        </content>
      </subnet>
      <subnet id="N2">
        <interface id="N2-interface">
          <gate action="input" id="N2-igp1" type="place"/>
          <gate action="output" id="N2-ogt1" type="transition"/>
        </interface>
        <content id="N2_content">
          <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
              Type="http://www.w3.org/2001/04/xmlenc#Content">
            <xenc:EncryptionMethod
                Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <xenc:EncryptedKey xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                <xenc:EncryptionMethod
                    Algorithm="http://www.w3.org/2001/04/xmlenc#kw-tripledes"/>
                <xenc:CipherData>
                  <xenc:CipherValue>
                    adSAmnteqXFUBdX3I6ClLoe27subXx/IPie9RUh3Viw=
                  </xenc:CipherValue>
                </xenc:CipherData>
              </xenc:EncryptedKey>
            </ds:KeyInfo>
            <xenc:CipherData>
              <xenc:CipherValue>
f3qOP5rBe3h/B4j1ruwdyBHDf59hu1Z27UhN9U4VgUisgzKcFcT6dO6GGtveXBoSHKv/oi8ZVCsS
9XWuAVZsMk7LB4M2A3hKsDqEpGsPWD1S+9Z84vNZbq1YGaukA/VqeCV/tmTEIS9p5ygFJwszS9Ri
o81rQ+dyOB9GsyLl1HQbaeRAvbY5mut5z+wdgOeiBmdTLkbJ+WGqvORNK/X8lxGmTdqNiczWvHQ7
cozZbJo0nG2UqtUqLRNnLIfHL3YTORLpEEwg09rUIgxHr9eNv/Q7Nxxi4Rv3DeGL9LGWoZm2F8a6
rUQ8vxoQBPzTkvcAFcMyD2VWx3YfirCE+CsIW1rNEm8YKRsbix4TDBjkAzU7shnHtlCR+AwSF24V
```

```
TtqMSZEvdDBoTYaublqqwpuwjejIyJ72oswQgKDPYdgVNZrtUL5Cf7VLXVzD7gTlg2rRFaEPUlID
jDQiHE7JHNmKWlk4MaJ/XKnQ/uyEdFNF7BDpykwZhWkDZsM+rXSauG72bmtBc6xljdrllwMtE9pV
Q/T4PO6Q3zORLQ6+vTlYDCQ9N0AsPNt481I2cOg/T/qOteTk2/vavoU8fO2e/RKDjLSzznRe52aV
GTfbLnbjNgewPcIFp8qFUPhWBExfs2Pa
              </xenc:CipherValue>
            </xenc:CipherData>
          </xenc:EncryptedData>
        </content>
      </subnet>
      <subnet id="N3">
        <interface id="N3-interface">
          <gate action="input" id="N3-igp1" type="place"/>
          <gate action="output" id="N3-ogt1" type="transition"/>
        </interface>
        <content id="N3_content">
          <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
              Type="http://www.w3.org/2001/04/xmlenc#Content">
            <xenc:EncryptionMethod
                Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <xenc:EncryptedKey
                  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                <xenc:EncryptionMethod
                    Algorithm="http://www.w3.org/2001/04/xmlenc#kw-tripledes"/>
                <xenc:CipherData>
                  <xenc:CipherValue>
                    adSAmnteqXFUBdX3I6ClLoe27subXx/IPie9RUh3Viw=
                  </xenc:CipherValue>
                </xenc:CipherData>
              </xenc:EncryptedKey>
            </ds:KeyInfo>
            <xenc:CipherData>
              <xenc:CipherValue>
0BQYbCoHJjFPwKoFK6WbZ/0z0GT16QoHSJnJ8S2KSuKaXrbKrX7i+mAEXWqPIkBi0FZg18FwHlba
BD2lD3B+M00Je+Jvig1r7rxwVtI4ZkCDlDBi7cC0uzB3E6f3WmD1Rz8PyigMfAwkUW8bHnblqU+R
SzRZEyjt9B8NEe3rCtuEEQRs/HGa/WTrYP9wUjx0KKYkLTD5PUz3tAsHFTpFOTlw9jMgIq5QC9eP
wuufPGQ=
              </xenc:CipherValue>
            </xenc:CipherData>
          </xenc:EncryptedData>
        </content>
      </subnet>
      <place id="p1"/>
      <place id="p6"/>
      <transition id="t4"/>
      <transition id="t8"/>
      <arc id="a1" source="p1" target="N1-igp1"/>
      <arc id="a7" source="N1-ogt1" target="N3-igp1"/>
      <arc id="a11" source="N1-ogp1" target="p1"/>
      <arc id="a12" source="N3-ogt1" target="N2-igp1"/>
      <arc id="a16" source="N2-ogt1" target="t8"/>
      <arc id="a17" source="t8" target="p1"/>
    </page>
  </net>
</pnml>
```

Now, suppose that we have other subnet ciphered with the same interface as N1. For example:


```
<subnet id="N5">
  <interface id="N5-interface">
    <gate action="input" id="N5-igp1" type="place"/>
    <gate action="output" id="N5-ogt1" type="transition"/>
    <gate action="output" id="N5-ogp1" type="place"/>
  </interface>
  <content id="N5_content">
    <xenc:EncryptedData
        xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
        Type="http://www.w3.org/2001/04/xmlenc#Content">
      <xenc:EncryptionMethod
          Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <xenc:EncryptedKey xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
          <xenc:EncryptionMethod
              Algorithm="http://www.w3.org/2001/04/xmlenc#kw-tripledes"/>
          <xenc:CipherData>
            <xenc:CipherValue>
              7fw3o2IsAjwMotze2QUnYLQKm0KawfqhEMd2pLUGvd8=
            </xenc:CipherValue>
          </xenc:CipherData>
        </xenc:EncryptedKey>
      </ds:KeyInfo>
      <xenc:CipherData>
        <xenc:CipherValue>
iRTrqnS0l66bI0R71gU6wY8kASLqCsF43Ljbpe72Ne2JDENTBp1I6oapcgxr8pLfDD2XpzFKV9cE
EXyBHcuS2MG2YVrf0QVoHNIvp7y0246bWei6HlXPZH4sGV0tlSZKqlU5kWtV3k++ZQ9yyJ0F7pG1
KdqCa0GJ0j64wpf/U0+KQeDVhVc9toD4vj9HxGqAMa6pFizHmqg0B/iNjz6fFXN7JFkevjx6a/e8
QPHlpR/bLR/43P4Q7lWOJt9FTU/GAd3bOoIOY3qMhN3hgLTcWv47Uh6gymAgnXhAqygBNcYyHS79
4fXjQUh8DFW2AlE6Bx4hHC/tNd+shRGbaZZ4hmLqVhSe7DWB3px9WDtpN2JMeDGen3TvSSNPjAnk
h5lkchzCajgFRCAAxxHMmm9BtVBdQ+4ZejKBJKas8kXcDwc2hYQEnK31p2SXNMmCJwpTSoiNIR7s
9YS6R6r4IJlmvR/hlKxGF/ZGxAFBtFRjsBoN5aI03L6wwmOfxOgl2sNvpjpvU3xYb7+fFP9s1eJ5
pJW6ZGMAegsUd+VPDq+QolcLXs9jBW6ADTcPP3ZgLDX75+PTTVOqnYYrQhLz5fqDC9iwzWzI68bc
moUqy7yKq8wDdtlSh0TPMYg33O5u6S9UcI7Pc0bElRu8rLS0JU+eAt/+v7PfXXLqM/J1Oux9MGcJ
eivcx9OTX3xx6BTjRee6GrNy0/jiv51Q8MWncWfbu+lPhmYyWtWJAOC8ixXHQpYNHVoFJm1QOOaA
KolYfVbgIsS1868HGkTPtAb5/8pJIyi8X2h1VmHPbbxwjAmOA/zNEWjY+uv2x/lkZ3BN86wcsvW2
4LaKU/kaiAasPANHazl6RzoTqEtBMO6E/ihwmpHaQSOJebG0YzexJ4v66PxyCgPMzEoz8xtwQeQs
7d4Nrtebhj5ZiAWdtDi2uwt/6i9RzOLdXSQC10pqDALSYlQQYa1hLm8+li4a9xkkA3jcuqYXaIku
d88wfI8lSeLzCSyJfTZYORtPrn5QwxcyNQulrIDJlQgyY2r/CENKqIcN5nAuOQnwEQx5GvihSEEk
aV1gy1Ji1GDaQkMbNAVJmkMZZWlhUC3+ODMChKT1r6jH7UO0lylOnZCOiWL7j9P9wbsN1WR+49CE
yKOi7HNSjqzIk71U7Z2HiIOCAOmYOT5Lb6j2K19xVYH5wYqqV5acqqKSz7/2wDY4NycfB56y2Uor
uO5YakMbNM6TWvOIi6hkTsF83gI4cEfPjw4t18fRw1eb9ULJ7Qk0o3S64FvbUuxYQrgjHg/MfM9d
WL0BzHiDdw==
        </xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </content>
</subnet>
```


As the encryption only affect to the replaced content, I can take this subnet and replace it in the other net, instead of N1. The only thing I have to do is to change the origin or

target of the arcs. Normally, the gates ids are different and this has to be reflected in the code. So this PNML content is valid too:


```
<?xml version="1.0" encoding="UTF-8"?>
<pnml xmlns="http://www.pnml.org/version-2009/grammar/pnml">
  <net id="latorre1" type="http://www.pnml.org/version-2009/grammar/ptnet">
    <page id="page1">
      <subnet id="N5">
        <interface id="N5-interface">
          <gate action="input" id="N5-igp1" type="place"/>
          <gate action="output" id="N5-ogt1" type="transition"/>
          <gate action="output" id="N5-ogp1" type="place"/>
        </interface>
        <content id="N5_content">
          <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#" Type="http://www.w3
            <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <xenc:EncryptedKey xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#kw-tripledes"
                <xenc:CipherData>
                  <xenc:CipherValue>
              7fw3o2IsAjwMotze2QUnYLQKm0KawfqhEMd2pLUGvd8=
            </xenc:CipherValue>
                </xenc:CipherData>
              </xenc:EncryptedKey>
            </ds:KeyInfo>
            <xenc:CipherData>
              <xenc:CipherValue>
iRTrqnS0l66bI0R71gU6wY8kASLqCsF43Ljbpe72Ne2JDENTBp1I6oapcgxr8pLfDD2XpzFKV9cE
EXyBHcuS2MG2YVrf0QVoHNIvp7y0246bWei6HlXPZH4sGVOtlSZKqlU5kWtV3k++ZQ9yyJ0F7pG1
KdqCaOGJ0j64wpf/UO+KQeDVhVc9toD4vj9HxGqAMa6pFizHmqg0B/iNjz6fFXN7JFkevjx6a/e8
QPHlpR/bLR/43P4Q7lWOJt9FTU/GAd3bOoIOY3qMhN3hgLTcWv47Uh6gymAgnXhAqygBNcYyHS79
4fXjQUh8DFW2AlE6Bx4hHC/tNd+shRGbaZZ4hmLqVhSe7DWB3px9WDtpN2JMeDGen3TvSSNPjAnk
h5lkchzCajgFRCAAxxHMmm9BtVBdQ+4ZejKBJKas8kXcDwc2hYQEnK31p2SXNMmCJwpTSoiNIR7s
9YS6R6r4IJlmvR/hlKxGF/ZGxAFBtFRjsBoN5aI03L6wwmOfxOgl2sNvpjpvU3xYb7+fFP9s1eJ5
pJW6ZGMAegsUd+VPDq+QolcLXs9jBW6ADTcPP3ZgLDX75+PTTVOqnYYrQhLz5fqDC9iwzWzI68bc
moUqy7yKq8wDdtlSh0TPMYg33O5u6S9UcI7Pc0bElRu8rLS0JU+eAt/+v7PfXXLqM/J1Oux9MGcJ
eivcx9OTX3xx6BTjRee6GrNy0/jiv51Q8MWncWfbu+lPhmYyWtWJAOC8ixXHQpYNHVoFJm1QOOaA
KolYfVbgIsS1868HGkTPtAb5/8pJIyi8X2h1VmHPbbxwjAmOA/zNEWjY+uv2x/lkZ3BN86wcsvW2
4LaKU/kaiAasPANHazl6RzoTqEtBMO6E/ihwmpHaQSOJebG0YzexJ4v66PxyCgPMzEoz8xtwQeQs
7d4Nrtebhj5ZiAWdtDi2uwt/6i9RzOLdXSQC10pqDALSYlQQYa1hLm8+li4a9xkkA3jcuqYXaIku
d88wfI8lSeLzCSyJfTZYORtPrn5QwxcyNQulrIDJlQgyY2r/CENKqIcN5nAuOQnwEQx5GvihSEEk
aV1gy1Ji1GDaQkMbNAVJmkMZZWlhUC3+ODMChKT1r6jH7UOOlylOnZCOiWL7j9P9wbsN1WR+49CE
yKOi7HNSjqzIk71U7Z2HiIOCAOmYOT5Lb6j2K19xVYH5wYqqV5acqqKSz7/2wDY4NycfB56y2Uor
uO5YakMbNM6TWvOIi6hkTsF83gI4cEfPjw4t18fRw1eb9ULJ7Qk0o3S64FvbUuxYQrgjHg/MfM9d
WL0BzHiDdw==
            </xenc:CipherValue>
                </xenc:CipherData>
          </xenc:EncryptedData>
        </content>
      </subnet>
      <subnet id="N2">
        <interface id="N2-interface">
          <gate action="input" id="N2-igp1" type="place"/>
          <gate action="output" id="N2-ogt1" type="transition"/>
```

```
        </interface>
        <content id="N2_content">
          <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#" Type="http://www.w3
            <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <xenc:EncryptedKey xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#kw-tripledes"
                <xenc:CipherData>
                  <xenc:CipherValue>
                    adSAmnteqXFUBdX3I6ClLoe27subXx/IPie9RUh3Viw=
                  </xenc:CipherValue>
                </xenc:CipherData>
              </xenc:EncryptedKey>
            </ds:KeyInfo>
            <xenc:CipherData>
              <xenc:CipherValue>
f3qOP5rBe3h/B4j1ruwdyBHDf59hu1Z27UhN9U4VgUisgzKcFcT6dO6GGtveXBoSHKv/oi8ZVCsS
9XWuAVZsMk7LB4M2A3hKsDqEpGsPWD1S+9Z84vNZbq1YGaukA/VqeCV/tmTEIS9p5ygFJwszS9Ri
o81rQ+dyOB9GsyLl1HQbaeRAvbY5mut5z+wdgOeiBmdTLkbJ+WGqv0RNK/X8lxGmTdqNiczWvHQ7
cozZbJo0nG2UqtUqLRNnLIfHL3YTORLpEEwgO9rUIgxHr9eNv/Q7Nxxi4Rv3DeGL9LGWoZm2F8a6
rUQ8vxoQBPzTkvcAFcMyD2VWx3YfirCE+CsIW1rNEm8YKRsbix4TDBjkAzU7shnHtlCR+AwSF24V
TtqMSZEvdDBoTYaublqqwpuwjejIyJ72oswQgKDPYdgVNZrtUL5Cf7VLXVzD7gTlg2rRFaEPUlID
jDQiHE7JHNmKWlk4MaJ/XKnQ/uyEdFNF7BDpykwZhWkDZsM+rXSauG72bmtBc6xljdrllwMtE9pV
Q/T4PO6Q3zORLQ6+vTlYDCQ9N0AsPNt481I2cOg/T/qOteTk2/vavoU8fO2e/RKDjLSzznRe52aV
GTfbLnbjNgewPcIFp8qFUPhWBExfs2Pa
              </xenc:CipherValue>
            </xenc:CipherData>
          </xenc:EncryptedData>
        </content>
      </subnet>
      <subnet id="N3">
        <interface id="N3-interface">
          <gate action="input" id="N3-igp1" type="place"/>
          <gate action="output" id="N3-ogt1" type="transition"/>
        </interface>
        <content id="N3_content">
          <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#" Type="http://www.w3
            <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"/>
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <xenc:EncryptedKey xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#kw-tripledes"
                <xenc:CipherData>
                  <xenc:CipherValue>
                    adSAmnteqXFUBdX3I6ClLoe27subXx/IPie9RUh3Viw=
                  </xenc:CipherValue>
                </xenc:CipherData>
              </xenc:EncryptedKey>
            </ds:KeyInfo>
            <xenc:CipherData>
              <xenc:CipherValue>
0BQYbCoHJjFPwKoFK6WbZ/0z0GT16QoHSJnJ8S2KSuKaXrbKrX7i+mAEXWqPIkBi0FZg18FwHlba
BD2lD3B+M00Je+Jvig1r7rxwVtI4ZkCDlDBi7cC0uzB3E6f3WmD1Rz8PyigMfAwkUW8bHnblqU+R
SzRZEyjt9B8NEe3rCtuEEQRs/HGa/WTrYP9wUjx0KKYkLTD5PUz3tAsHFTpFOTlw9jMgIq5QC9eP
wuufPGQ=
              </xenc:CipherValue>
```

```
            </xenc:CipherData>
          </xenc:EncryptedData>
        </content>
      </subnet>
      <place id="p1"/>
      <place id="p6"/>
      <transition id="t4"/>
      <transition id="t8"/>
      <arc id="a1" source="p1" target="N5-igp1"/>
      <arc id="a7" source="N5-ogt1" target="N3-igp1"/>
      <arc id="a11" source="N5-ogp1" target="p1"/>
      <arc id="a12" source="N3-ogt1" target="N2-igp1"/>
      <arc id="a16" source="N2-ogt1" target="t8"/>
      <arc id="a17" source="t8" target="p1"/>
    </page>
  </net>
</pnml>
```

5.3 XMLSignature

5.3.1 Introduction

Although privacy is a solved question with XMLEncryption, there are other aspects of the security that XMLEncryption can't cover. For example:

- We want nobody to modify parts of a Petri net.
- I want to put on record that I am the Petri net author or responsible.
- Maybe I want to be sure that the author of a Petri net is somebody with no doubt.

To solve these questions, we can use digital signature. It gives us several interesting characteristics:

- **Integrity**: The data integrity will appear if we are able to avoid, or at least detect, any non authorized modification of the information
- **Authentication**: authentication guarantee that somebody is who says that he is. Basically, with digital signature nobody can impersonate another.
- **Non repudiation**: with this characteristic, we can avoid that somebody says he hasn't signed something if he really did. Only that person can have done it.

In this case I want to sign a whole Petri net or concrete parts of it. Obviously, if I sign only a fragment of a Petri net, this part keeps integrity, authentication and non repudiation, but the rest of net doesn't.

As with XMLEncryption, the best way to achieve this goals is using standard technologies. For signing, the method chosen is XMLSignature [76].

5.3.2 XMLSignature revision

XMLSignature is a digital signing standard. With XMLSignature we can sign any kind of content but the result is XML content. It requires the use of digital certificates and a set of public/private keys, using asymmetrical ciphering algorithms for the process.

It has three possibilities:

1. **Enveloped**: The result of the signing is the original xml file with a new attached signature element inside.
2. **Enveloping**: The result of the signing is a new xml file with the digital signature, and, inside it, the original signed elements of the initial xml file.
3. **Detached**: The result of the signature is a new independent file. The original xml file remains inalterable, The new file contains the sign.

It is indifferent which of this three methods use. They are only different ways to organize the generated signature.

XMLSignature forces a digital signature to have:

- **Canonicalization method**: Basically, we have an equivalent relationship that is able to detect if two xml files are equivalent each other. A canonicalization method transforms a xml file into a canonicalized one that is the representant of the equivalence class. All the equivalent xml files are transformed in this representant. This transform has to be applied before signing. If no method is declared, there is one defined by default.
- **Reference**: there can be several references in one only signature. Each reference indicates a part of the document that has to be signed and the digest algorithm used[1]. It is important to say that each reference doesn't generate a signature, but all of these references are signed together and generates only one signature.

[1] A digest algorithm generates a fixed length bytes sequence from arbitrary length contents. This bytes sequence is different for each content

- **Key information**: Optionally, the signature can include necessary information to be validated. In this part we can indicate the public key directly, by a bytes sequence, by an URL or the most usual X.509 digital certificates.

- **Transforms**: It is possible that what we want to sign is not the whole document, but concrete information obtained from it (but not the content itself): e.g. encoding/decoding, select only certain parts, modify the XML structure, include fragments of other documents,... Transforms is a ordered list of processing steps that has to be applied to the content before being digested. This feature is optional but in this case it will be necessary.

A XML signature consists of a tag `<Signature>` that is defined in the following namespace: `http://www.w3.org/2000/09/xmldsig#`

The basic structure of the element signature is:

```
<Signature>
  <SignedInfo>
    <CanonicalizationMethod />
    <SignatureMethod />
    <Reference>
      <Transforms>
        <Transform />
      </Transforms>
      <DigestMethod>
      <DigestValue>
    </Reference>
    <Reference /> etc.
  </SignedInfo>
  <SignatureValue />
  <KeyInfo />
  <Object />
</Signature>
```

The element `<Object>` is used only in enveloping signature to store the signed data. In this case I am going to use enveloped signature, so this element will no be present in the examples.

The final aspect of a complete XMLSignature element is like this example:

```
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
  <SignedInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
    <CanonicalizationMethod
        Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"
        xmlns="http://www.w3.org/2000/09/xmldsig#" />
    <SignatureMethod
        Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1"
        xmlns="http://www.w3.org/2000/09/xmldsig#" />
    <Reference URI=""
        xmlns="http://www.w3.org/2000/09/xmldsig#">
```

```
      <Transforms
          xmlns="http://www.w3.org/2000/09/xmldsig#">
        <Transform
            Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"
            xmlns="http://www.w3.org/2000/09/xmldsig#" />
      </Transforms>
      <DigestMethod
          Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"
          xmlns="http://www.w3.org/2000/09/xmldsig#" />
      <DigestValue
          xmlns="http://www.w3.org/2000/09/xmldsig#">
        0yyx+K28+cp7kuUgcnANtTBdUwg=
      </DigestValue>
    </Reference>
  </SignedInfo>
  <SignatureValue xmlns="http://www.w3.org/2000/09/xmldsig#">
    ZVzRud7G4mEZsDnBavbnZoFUmm5J2OBDkQ+IooDLn95ndGYdrq6uPQ==
  </SignatureValue>
  <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
    <X509Data xmlns="http://www.w3.org/2000/09/xmldsig#">
      <X509Certificate
          xmlns="http://www.w3.org/2000/09/xmldsig#">
        IICmDCCAlYCBEfrim8wCwYHKoZIzjgEAwUAMDIxCzAJBgNVBAYTAkVTMREwDwYDVQQKEwhBd
        pYTEQMA4GA1UEAxMHVXN1YXJpbzAeFw0wODAzMjcxMTUyMTVaFw0wOTAzMjcxMTUyMTVaMDI
        BgNVBAYTAkVTMREwDwYDVQQKEwhBdXRlbnRpYTEQMA4GA1UEAxMHVXN1YXJpbzCCAbcwggEs
        BgcqhkjOOAQBMIIBHwKBgQD9f1OBHXUSKVLfSpwu7OTn9hG3UjzvRADDHj+AtlEmaUVdQCJR
        jVj6v8X1ujD2y5tVbNeBO4AdNG/yZmC3a5lQpaSfn+gEexAiwk+7qdf+t8Yb+DtX58aophUP
        9tPFHsMCNVQTWhaRMvZ1864rYdcq7/IiAxmd0UgBxwIVAJdgUI8VIwvMspK5gqLrhAvwWBz1
        APfhoIXWmz3ey7yrXDa4V7l5lK+7+jrqgvlXTAs9B4JnUVlXjrrUWU/mcQcQgYC0SRZxI+hM
        t88JMozIpuE8FnqLVHyNKOCjrh4rs6Z1kW6jfwv6ITVi8ftiegEkO8yk8b6oUZCJqIPf4Vrl
        i2ZegHtVJWQBTDv+z0kqA4GEAAKBgDUPDwxDZFXMrZha74VNmgyFslLM01wKw17nbt9UFTJA
        iPpozeZMP2u0SoYst2nbxkCs1hziiuaNJnykzcjVf3+PmL3sQES8SxwJBRUME2UTA2006WD3
        iZ9yibcWQimB8eKIJyBBxSk5TueAzvTA8HN2+Rvgh8RMaOzhMAsGByqGSM44BAMFAAMvADAs
        4+nQZdFvlvsfyOfq1t02h9MJEgIUEvYDfxeygKCmrIlA0sQLtaCs0Qo=
      </X509Certificate>
    </X509Data>
    <KeyValue xmlns="http://www.w3.org/2000/09/xmldsig#">
      <DSAKeyValue
          xmlns="http://www.w3.org/2000/09/xmldsig#">
        <P xmlns="http://www.w3.org/2000/09/xmldsig#">
          /X9TgR11EilS30qcLuzk5/YRt1I870QAwx4/gLZRJmlFXUAiUftZPY1Y+r/F9bow9subVWz
          HTRv8mZgt2uZUKWkn5/oBHsQIsJPu6nX/rfGG/g7V+fGqKYVDwT7g/bTxR7DAjVUE1oWkTL
          K2HXKu/yIgMZndFIAcc=
        </P>
        <Q xmlns="http://www.w3.org/2000/09/xmldsig#">
          l2BQjxUjC8yykrmCouuEC/BYHPU=
        </Q>
        <G xmlns="http://www.w3.org/2000/09/xmldsig#">
          9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT+ZxBxCBgLRJFnEj6Ew
          zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx+2J6ASQ7zKTxvqhRkImog9/hWuW
          Zl6Ae1UlZAFMO/7PSSo=
        </G>
        <Y xmlns="http://www.w3.org/2000/09/xmldsig#">
          NQ8PDENkVcytmFrvhU2aDIWyUszTXArDXudu31QVMkAuTvWI+mjN5kw/a7RKhiy3advGQKz
          5o0mfKTNyNV/f4+YvexARLxLHAkFFQwTZRMDbTTpYPfE3L2Jn3KJtxZCKYHx4ognIEHFKTl
```

```
          9MDwc3b5G+CHxExo7OE=
        </Y>
      </DSAKeyValue>
    </KeyValue>
  </KeyInfo>
</Signature>
```

At first sight it seems very complicated, but it isn't. First of all, the attribute xmlns that is in almost all the tags is only the namespace. From here, for more clarity it can be obviated. So the example is as follows:

```
<Signature>
  <SignedInfo>
    <CanonicalizationMethod
        Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
    <SignatureMethod
        Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1" />
    <Reference URI="">
      <Transforms>
        <Transform
            Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
      </Transforms>
      <DigestMethod
          Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
      <DigestValue>
        Oyyx+K28+cp7kuUgcnANtTBdUwg=
      </DigestValue>
    </Reference>
  </SignedInfo>
  <SignatureValue>
    ZVzRud7G4mEZsDnBavbnZoFUmm5J2OBDkQ+IooDLn95ndGYdrq6uPQ==
  </SignatureValue>
  <KeyInfo>
    <X509Data>
      <X509Certificate>
        IICmDCCAlYCBEfrim8wCwYHKoZIzjgEAwUAMDIxCzAJBgNVBAYTAkVTMREwDwYDVQQKEwhBd
        pYTEQMA4GA1UEAxMHVXN1YXJpbzAeFw0wODAzMjcxMTUyMTVaFw0wOTAzMjcxMTUyMTVaMDI
        BgNVBAYTAkVTMREwDwYDVQQKEwhBdXRlbnRpYTEQMA4GA1UEAxMHVXN1YXJpbzCCAbcwggEs
        BgcqhkjOOAQBMIIBHwKBgQD9f1OBHXUSKVLfSpwu7OTn9hG3UjzvRADDHj+AtlEmaUVdQCJR
        jVj6v8X1ujD2y5tVbNeBO4AdNG/yZmC3a5lQpaSfn+gEexAiwk+7qdf+t8Yb+DtX58aophUP
        9tPFHsMCNVQTWhaRMvZ1864rYdcq7/IiAxmd0UgBxwIVAJdgUI8VIwvMspK5gqLrhAvwWBz1
        APfhoIXWmz3ey7yrXDa4V7l5lK+7+jrqgvlXTAs9B4JnUVlXjrrUWU/mcQcQgYC0SRZxI+hM
        t88JMozIpuE8FnqLVHyNKOCjrh4rs6Z1kW6jfwv6ITVi8ftiegEkO8yk8b6oUZCJqIPf4Vrl
        i2ZegHtVJWQBTDv+z0kqA4GEAAKBgDUPDwxDZFXMrZha74VNmgyFslLM01wKw17nbt9UFTJA
        iPpozeZMP2u0SoYst2nbxkCs1hziiuaNJnykzcjVf3+PmL3sQES8SxwJBRUME2UTA2006WD3
        iZ9yibcWQimB8eKIJyBBxSk5TueAzvTA8HN2+Rvgh8RMaOzhMAsGByqGSM44BAMFAAMvADAs
        4+nQZdFvlvsfyOfq1t02h9MJEgIUEvYDfxeygKCmrIlA0sQLtaCs0Qo=
      </X509Certificate>
    </X509Data>
    <KeyValue>
      <DSAKeyValue>
        <P>
          /X9TgR11EilS30qcLuzk5/YRt1I870QAwx4/gLZRJmlFXUAiUftZPY1Y+r/F9bow9subVWz
          HTRv8mZgt2uZUKWkn5/oBHsQIsJPu6nX/rfGG/g7V+fGqKYVDwT7g/bTxR7DAjVUE1oWkTL
```

```
          K2HXKu/yIgMZndFIAcc=
        </P>
        <Q>
          l2BQjxUjC8yykrmCouuEC/BYHPU=
        </Q>
        <G>
          9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT+ZxBxCBgLRJFnEj6Ew
          zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx+2J6ASQ7zKTxvqhRkImog9/hWuW
          Zl6Ae1UlZAFMO/7PSSo=
        </G>
        <Y>
          NQ8PDENkVcytmFrvhU2aDIWyUszTXArDXudu31QVMkAuTvWI+mjN5kw/a7RKhiy3advGQKz
          5o0mfKTNyNV/f4+YvexARLxLHAkFFQwTZRMDbTTpYPfE3L2Jn3KJtxZCKYHx4ognIEHFKTl
          9MDwc3b5G+CHxExo7OE=
        </Y>
      </DSAKeyValue>
    </KeyValue>
  </KeyInfo>
</Signature>
```


5.3.3 XMLSignature and Petri nets

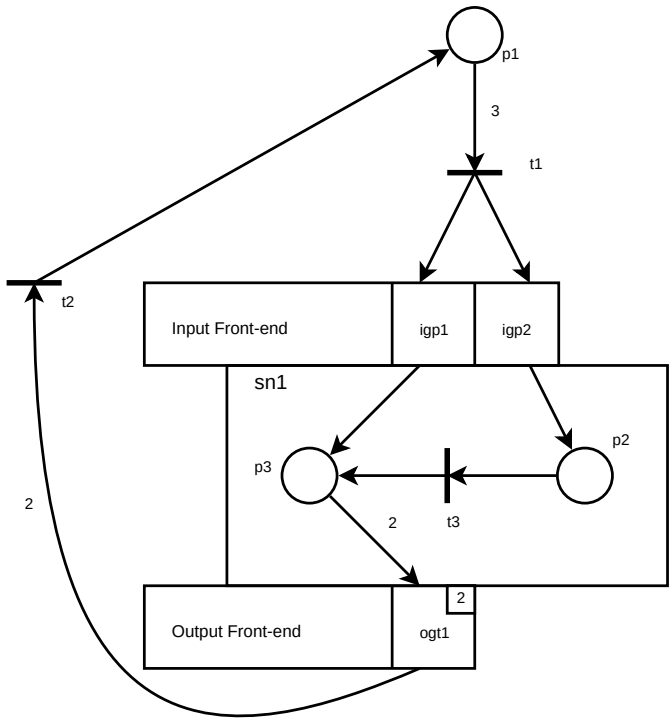
In this case I am going to use enveloped signature, so the result of the signature is stored in a new tag inside the original PNML file. As I explained before, we have to select which parts of the PNML file are going to be signed.

Many times we will need to sign the whole Petri net, but it will be very usual to sign only certain parts of a Petri net, for example a critical subprocess. The modus operandi here is similar to XMLEncryption. First of all, the content to be signed should be grouped in a subnet and then, this subnet is signed.

The standard way to indicate a subnet to sign in XMLSignature is through a XPath [82] expression. In XMLSignature, the way to specify XPath addresses is using XMLSignature XPath Filter [83]. XPathFilter returns the node set that is going to be signed and it is placed into `/Signature/SignedInfo/Reference/Transforms` as a new `<Transform>`.

I am not going to explain all the possibilities of XPath Filter. I will explain only those main configurations useful to my objective. The exact configuration depends on the particular necessities of each case.

In order to illustrate the process, let's take the figure 4.6 as example again:



and its full PNML representation:

```
<?xml version="1.0" encoding="utf-8"?>
<pnml>
  <net id="myNet" type="http://www.pnml.org/version-2009/grammar/ptnet">
    <name>
      <text> My new net </text>
    </name>
    <page id="page1">
      <subnet id="sn1">
        <interface id="sn1-interface">
        <gate id="igp1" action="input" type="place"/>
        <gate id="igp2" action="input" type="place"/>
        <gate id="ogt1" action="output" type="transition">
          <inscription>
          <text> 2 </text>
          </inscription>
        </gate>
        </interface>
        <content id="sn1-content">
        <place id="p2"/>
        <place id="p3"/>
        <transition id="t3"/>
        <arc id="sn1-a2" source="igp2" target="p2"/>
        <arc id="sn1-a3" source="igp1" target="p3"/>
        <arc id="sn1-a4" source="p3" target="ogt1">
          <inscription>
          <text> 2 </text>
          </inscription>
```

```
        </arc>
        <arc id="a5" source="t3" target="p3"/>
        <arc id="a6" source="p2" target="t3"/>
        </content>
      </subnet>
      <place id="p1"/>
      <transition id="t1"/>
      <transition id="t2"/>
      <arc id="a1" source="p1" target="t1">
        <inscription>
        <text> 3 </text>
        </inscription>
      </arc>
      <arc id="a2" source="t1" target="igp2"/>
      <arc id="a3" source="t1" target="igp1"/>
      <arc id="a4" source="ogt1" target="t2">
        <inscription>
        <text> 2 </text>
        </inscription>
      </arc>
      <arc id="a7" source="t2" target="p1"/>
    </page>
  </net>
</pnml>
```

The first option is to sign the whole net. In XPath, the expression to represent the entire document is:

```
/
```

If we apply XMLSignature with this XPath expression, the result is

```
<?xml version="1.0" encoding="UTF-8"?>
<pnml>
  <net id="myNet" type="http://www.pnml.org/version-2009/grammar/ptnet">
    <name>
      <text> My new net </text>
    </name>
    <page id="page1">
      <subnet id="sn1">
        <interface id="sn1-interface">
          <gate action="input" id="igp1" type="place"/>
          <gate action="input" id="igp2" type="place"/>
          <gate action="output" id="ogt1" type="transition">
            <inscription>
              <text> 2 </text>
            </inscription>
          </gate>
        </interface>
        <content id="sn1-content">
          <place id="p2"/>
          <place id="p3"/>
          <transition id="t3"/>
          <arc id="a2" source="igp2" target="p2"/>
          <arc id="a3" source="igp1" target="p3"/>
          <arc id="a4" source="p3" target="ogt1">
            <inscription>
              <text> 2 </text>
            </inscription>
          </arc>
```

```
          <arc id="a5" source="t3" target="p3"/>
          <arc id="a6" source="p2" target="t3"/>
        </content>
      </subnet>
      <place id="p1"/>
      <transition id="t1"/>
      <transition id="t2"/>
      <arc id="a1" source="p1" target="t1">
        <inscription>
          <text> 3 </text>
        </inscription>
      </arc>
      <arc id="a2" source="t1" target="igp2"/>
      <arc id="a3" source="t1" target="igp1"/>
      <arc id="a4" source="ogt1" target="t2">
        <inscription>
          <text> 2 </text>
        </inscription>
      </arc>
      <arc id="a7" source="t2" target="p1"/>
    </page>
  </net>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
      <ds:Reference URI="">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
          <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments"/>
          <ds:Transform Algorithm="http://www.w3.org/2002/06/xmldsig-filter2">
            <dsig-xpath:XPath xmlns:dsig-xpath="http://www.w3.org/2002/06/xmldsig-filter2" Filter="union">
              /
            </dsig-xpath:XPath>
          </ds:Transform>
        </ds:Transforms>
        <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <ds:DigestValue>LIMSPHwtGpK3hlDEdfaAv7D39KU=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>
      W3C5nlPSYcR2qvx2bOwpGy8CGMu0vWPTZCIQVkYuGUA52lRmtyFRInlRm0G+d+eqQHBwtvMophte
      HWPsPiOl+Bwc/C3HTHj2XBc9P1bcFUtVM91rLhLZhI/ZAl9t6VfLoQ+Cduu8sQJh6qiH24CiYGjc
      FaOlQbQOsYBvGXoBhEk=
    </ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:X509Certificate>
          MIICgTCCAeqgAwIBAgIETfh4CTANBgkqhkiG9w0BAQUFADCBhDELMAkGA1UEBhMCRVMxETAPBgNV
          BAgTCExBIFJJTOpBMREwDwYDVQQHDAhMT0dST8KlTzEgMB4GA1UEChMXVU5JVkVSUOlEQUQgREUg
          TEEgUklPSkExDDAKBgNVBAsTA1BGQzEfMBOGA1UEAwwWScKlSUdPIExFw6BOIFNBTUFOSUVHTzAe
          Fw0xMTA2MTUwOTEONDlaFw0xMTA5MTMwOTEONDlaMIGEMQswCQYDVQQGEwJFUzERMA8GA1UECBMI
          TEEgUklPSkExETAPBgNVBAcMCExPR1JPwqVPMSAwHgYDVQQKExdVTklWRVJTSURBRCBERSBMQSBS
          SU9KQTEMMAoGA1UECxMDUEZDMR8wHQYDVQQDDBZJwqVJRO8gTEXDoE4gUOFNQU5JRUdPMIGfMAOG
          CSqGSIb3DQEBAQUAA4GNADCBiQKBgQChePFNVCIfphFlyXQ9BysiR5BfXIuv3AnAK80Fuw4tTFwC
          nVUjJeGnkUYQO32oUu+fEBK8WsEqjeH8A7zrHTRQjfYZWyuGWrM8gJXOa/POMROPm/c/H8b5a6Nx
          1/+zLwROtYkqLI2xqDOFII2RwK5L2yGeV4T4y8i3h1UOOFTSEwIDAQABMAOGCSqGSIb3DQEBBQUA
          A4GBAIDOvAAdOCaTpy+83bGB2KmngMJrNxxWDpAi5LGFrN8iCShmbTpIeIbYBUAaBpZtdhOnhq4n
          wD5QOENSFipQcdH5GEpPM9Rquy6xMwfda9EU5UfOSEmbk4fK2vaIOVjynpQsJ9P99enO2smQlyvw
          /hBa7Xacz6qDut8ghUeuV5Js
        </ds:X509Certificate>
      </ds:X509Data>
      <ds:KeyValue>
        <ds:RSAKeyValue>
          <ds:Modulus>
            oXjxTVQiH6YRZclOPQcrIkeQX1yLr9wJwCvNBbsOLUxcAp1VIyXhp5FGEDt9qFLvnxASvFrBKo3h
            /AO86x00UI32GVsrhlqzPICVzmvz9DETj5v3Px/G+Wujcdf/sy8EdLWJKiyNsagzhSCNkcCuS9sh
            nleE+MvIt4dVNDhUOhM=
          </ds:Modulus>
          <ds:Exponent>AQAB</ds:Exponent>
        </ds:RSAKeyValue>
      </ds:KeyValue>
    </ds:KeyInfo>
  </ds:Signature>
</pnml>
```

In the generated tag `<Signature>` a XMLSignature XPath Filter tag has appeared as a new `<Transform>`

```
<ds:Transform Algorithm="http://www.w3.org/2002/06/xmldsig-filter2">
  <dsig-xpath:XPath
      xmlns:dsig-xpath="http://www.w3.org/2002/06/xmldsig-filter2"
      Filter="union">
    /
  </dsig-xpath:XPath>
</ds:Transform>
```

It is easy to see the behaviour of this element: the union of "/" XPath expression, that is to say, the union of the entire document.

The set of nodes returned by this expression is

```
<pnml>
  <net id="myNet" type="http://www.pnml.org/version-2009/grammar/ptnet">
    <name>
      <text> My new net </text>
    </name>
    <page id="page1">
      <subnet id="sn1">
        <interface id="sn1-interface">
        <gate action="input" id="igp1" type="place"></gate>
        <gate action="input" id="igp2" type="place"></gate>
        <gate action="output" id="ogt1" type="transition">
          <inscription>
          <text> 2 </text>
          </inscription>
        </gate>
        </interface>
        <content id="sn1-content">
        <place id="p2"></place>
        <place id="p3"></place>
        <transition id="t3"></transition>
        <arc id="sn1-a2" source="igp2" target="p2"></arc>
        <arc id="sn1-a3" source="igp1" target="p3"></arc>
        <arc id="sn1-a4" source="p3" target="ogt1">
          <inscription>
          <text> 2 </text>
          </inscription>
        </arc>
        <arc id="a5" source="t3" target="p3"></arc>
        <arc id="a6" source="p2" target="t3"></arc>
        </content>
      </subnet>
      <place id="p1"></place>
      <transition id="t1"></transition>
      <transition id="t2"></transition>
      <arc id="a1" source="p1" target="t1">
        <inscription>
```

```
        <text> 3 </text>
        </inscription>
      </arc>
      <arc id="a2" source="t1" target="igp2"></arc>
      <arc id="a3" source="t1" target="igp1"></arc>
      <arc id="a4" source="ogt1" target="t2">
        <inscription>
        <text> 2 </text>
        </inscription>
      </arc>
      <arc id="a7" source="t2" target="p1"></arc>
    </page>
  </net>
</pnml>
```

It is the full document (without the first row xml definition) node set.

Other important configuration in this work is the signing of a concrete subnet. In this case, it is a little different as in XMLEncryption. Remember that in XMLEncryption, if I want to mask a subnet I don't process the `<subnet>` tag but the `subnet/content`. This is because the interface has to be visible. But in a signature I want to sign the complete subnet, including the interface. Suppose that this subnet has `id="sn1"`. The XPath expression that represents it is:

```
/pnml/net/page/subnet[@id="sn1"]
```

The node set in this case is

```
<subnet id="sn1">
  <interface id="sn1-interface">
    <gate action="input" id="igp1" type="place"/>
    <gate action="input" id="igp2" type="place"/>
    <gate action="output" id="ogt1" type="transition">
      <inscription>
        <text> 2 </text>
      </inscription>
    </gate>
  </interface>
  <content id="sn1-content">
    <place id="p2"/>
    <place id="p3"/>
    <transition id="t3"/>
    <arc id="sn1-a2" source="igp2" target="p2"/>
    <arc id="sn1-a3" source="igp1" target="p3"/>
    <arc id="sn1-a4" source="p3" target="ogt1">
      <inscription>
        <text> 2 </text>
      </inscription>
    </arc>
    <arc id="a5" source="t3" target="p3"/>
    <arc id="a6" source="p2" target="t3"/>
  </content>
</subnet>
```

and the signature result is

```xml
<?xml version="1.0" encoding="UTF-8"?>
<pnml>
  <net id="myNet" type="http://www.pnml.org/version-2009/grammar/ptnet">
    <name>
      <text> My new net </text>
    </name>
    <page id="page1">
      <subnet id="sn1">
        <interface id="sn1-interface">
          <gate action="input" id="igp1" type="place"/>
          <gate action="input" id="igp2" type="place"/>
          <gate action="output" id="ogt1" type="transition">
            <inscription>
              <text> 2 </text>
            </inscription>
          </gate>
        </interface>
        <content id="sn1-content">
          <place id="p2"/>
          <place id="p3"/>
          <transition id="t3"/>
          <arc id="sn1-a2" source="igp2" target="p2"/>
          <arc id="sn1-a3" source="igp1" target="p3"/>
          <arc id="sn1-a4" source="p3" target="ogt1">
            <inscription>
              <text> 2 </text>
            </inscription>
          </arc>
          <arc id="a5" source="t3" target="p3"/>
          <arc id="a6" source="p2" target="t3"/>
        </content>
      </subnet>
      <place id="p1"/>
      <transition id="t1"/>
      <transition id="t2"/>
      <arc id="a1" source="p1" target="t1">
        <inscription>
          <text> 3 </text>
        </inscription>
      </arc>
      <arc id="a2" source="t1" target="igp2"/>
      <arc id="a3" source="t1" target="igp1"/>
      <arc id="a4" source="ogt1" target="t2">
        <inscription>
          <text> 2 </text>
        </inscription>
      </arc>
      <arc id="a7" source="t2" target="p1"/>
    </page>
  </net>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
      <ds:Reference URI="">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
          <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments"/>
          <ds:Transform Algorithm="http://www.w3.org/2002/06/xmldsig-filter2">
            <dsig-xpath:XPath xmlns:dsig-xpath="http://www.w3.org/2002/06/xmldsig-filter2" Filter="intersect">
              /pnml/net/page/subnet[@id="sn1"]
            </dsig-xpath:XPath>
          </ds:Transform>
        </ds:Transforms>
        <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <ds:DigestValue>prCzhLgTCZ1ck6MjQnFy6cASCZw=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>
      QoO7mQmGBFTg2UxgiZnzlsnKi8V477JC0v12JPItL53zIOCpjhOwLoyxENl6v8lCLoJ9WwFHlBKk
      r3GdqrgZimNXMUjwR4zkd9FVNcIrn85DuRjHA/zDwSuPMq9w0N5A07c0xJ24uvn9+zpbQxfblYTb
      kiy08+S0pqczU/bv5+g=
    </ds:SignatureValue>
    <ds:KeyInfo>
```

```
      <ds:X509Data>
        <ds:X509Certificate>
          MIICgTCCAeqgAwIBAgIETfh4CTANBgkqhkiG9w0BAQUFADCBhDELMAkGA1UEBhMCRVMxETAPBgNV
          BAgTCExBIFJJT0pBMREwDwYDVQQHDAhMT0dST8KlTzEgMB4GA1UEChMXVU5JVkVSU01EQUQgREUg
          TEEgUklPSkExDDAKBgNVBAsTA1BGQzEfMB0GA1UEAwwWScKlSUdPIExFw6BOIFNBTUFOSUVHTzAe
          Fw0xMTA2MTUw0TE0NDlaFw0xMTA5MTMw0TE0NDlaMIGEMQswCQYDVQQGEwJFUzERMA8GA1UECBMI
          TEEgUklPSkExETAPBgNVBAcMCExPR1JPwqVPMSAwHgYDVQQKExdVTklWRVJTSURBRCBERSBMQSBS
          SU9KQTEMMAoGA1UECxMDUEZDMR8wHQYDVQQDDBZJwqVJR08gTEXDoE4gU0FNQU5JRUdPMIGfMA0G
          CSqGSIb3DQEBAQUAA4GNADCBiQKBgQChePFNVCIfphFlyXQ9BysiR5BfXIuv3AnAK80Fuw4tTFwC
          nVUjJeGnkUYQO32oUu+fEBK8WsEqjeH8A7zrHTRQjfYZWyuGWrM8gJXOa/POMROPm/c/H8b5a6Nx
          1/+zLwR0tYkqLI2xqDOFII2RwK5L2yGeV4T4y8i3h1UOOFTSEwIDAQABMA0GCSqGSIb3DQEBBQUA
          A4GBAIDOvAAdOCaTpy+83bGB2KmngMJrNxxWDpAi5LGFrN8iCShmbTpIeIbYBUAaBpZtdhOnhq4n
          wD5QOENSFipQcdH5GEpPM9Rquy6xMwfda9EU5UfOSEmbk4fK2vaIOVjynpQsJ9P99enO2smQlyvw
          /hBa7Xacz6qDut8ghUeuV5Js
        </ds:X509Certificate>
      </ds:X509Data>
      <ds:KeyValue>
        <ds:RSAKeyValue>
          <ds:Modulus>
            oXjxTVQiH6YRZcl0PQcrIkeQX1yLr9wJwCvNBbs0LUxcAp1VIyXhp5FGEDt9qFLvnxASvFrBKo3h
            /AO86x00UI32GVsrhlqzPICVzmvz9DETj5v3Px/G+Wujcdf/sy8EdLWJKiyNsagzhSCNkcCuS9sh
            nleE+MvIt4dVNDhU0hM=
          </ds:Modulus>
          <ds:Exponent>AQAB</ds:Exponent>
        </ds:RSAKeyValue>
      </ds:KeyValue>
    </ds:KeyInfo>
  </ds:Signature>
</pnml>
```


In this case, the Transform associated to the XPath expression is

```
<ds:Transform Algorithm="http://www.w3.org/2002/06/xmldsig-filter2">
  <dsig-xpath:XPath
      xmlns:dsig-xpath="http://www.w3.org/2002/06/xmldsig-filter2"
      Filter="intersect">
    /pnml/net/page/subnet[@id="sn1"]
  </dsig-xpath:XPath>
</ds:Transform>
```

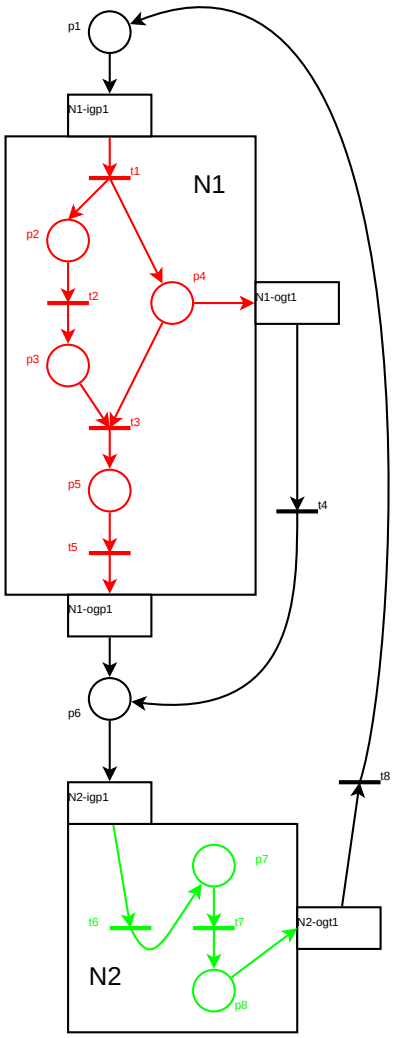
As we can see, the resultant node set is the intersection of the entire document and the nodes returned by `/pnml/net/page/subnet[@id="sn1"]`, that is, the nodes returned by `/pnml/net/page/subnet[@id="sn1"]`.

5.3.4 Example. Signing all the subnets of a Petri net

Other important configuration is to sign all the subnets defined in a PNML file. Let's take the example Petri net of the figure 4.8 again. The goal is to sign all the subnets, so we are going to use this XPath expression:

```
/pnml/net/page/subnet
```

Then we have the following Petri net



and this is the XML content with the signature of all the subnets:

```
<?xml version="1.0" encoding="UTF-8"?>
<pnml xmlns="http://www.pnml.org/version-2009/grammar/pnml">
  <net id="latorre1" type="http://www.pnml.org/version-2009/grammar/ptnet">
    <page id="page1">
      <subnet id="N1">
        <interface id="N1-interface">
          <gate action="input" id="N1-igp1" type="place"/>
          <gate action="output" id="N1-ogt1" type="transition"/>
          <gate action="output" id="N1-ogp1" type="place"/>
        </interface>
        <content id="N1_content">
          <place id="p2"/>
          <place id="p3"/>
          <place id="p4"/>
          <place id="p5"/>
          <transition id="t1"/>
          <transition id="t2"/>
          <transition id="t3"/>
          <transition id="t5"/>
          <arc id="a2" source="t1" target="p2"/>
          <arc id="a3" source="p2" target="t2"/>
          <arc id="a4" source="t2" target="p3"/>
          <arc id="a5" source="t1" target="p4"/>
          <arc id="a6" source="p3" target="t3"/>
          <arc id="a8" source="p4" target="t3"/>
          <arc id="a9" source="t3" target="p5"/>
          <arc id="a10" source="p5" target="t5"/>
```

```
        <arc id="N1-a1" source="N1-igp1" target="t1"/>
        <arc id="N1-a7" source="p4" target="N1-ogt1"/>
        <arc id="N1-a11" source="t5" target="N1-ogp1"/>
      </content>
    </subnet>
    <subnet id="N2">
      <interface id="N2-interface">
        <gate action="input" id="N2-igp1" type="place"/>
        <gate action="output" id="N2-ogt1" type="transition"/>
      </interface>
      <content id="N2_content">
        <place id="p7"/>
        <place id="p8"/>
        <transition id="t6"/>
        <transition id="t7"/>
        <arc id="a13" source="t6" target="p7"/>
        <arc id="a14" source="p7" target="t7"/>
        <arc id="a15" source="t7" target="p8"/>
        <arc id="N2-a12" source="N2-igp1" target="t6"/>
        <arc id="N2-a16" source="p8" target="N2-ogt1"/>
      </content>
    </subnet>
    <place id="p1"/>
    <place id="p6"/>
    <transition id="t4"/>
    <transition id="t8"/>
    <arc id="a1" source="p1" target="N1-igp1"/>
    <arc id="a7" source="N1-ogt1" target="t4"/>
    <arc id="a11" source="N1-ogp1" target="p6"/>
    <arc id="a12" source="p6" target="N2-igp1"/>
    <arc id="a16" source="N2-ogt1" target="t8"/>
    <arc id="a17" source="t8" target="p1"/>
    <arc id="a18" source="t4" target="p6"/>
  </page>
</net>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
    <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
    <ds:Reference URI="">
      <ds:Transforms>
        <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
        <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315#WithComments"/>
        <ds:Transform Algorithm="http://www.w3.org/2002/06/xmldsig-filter2">
          <dsig-xpath:XPath xmlns:dsig-xpath="http://www.w3.org/2002/06/xmldsig-filter2"
              Filter="intersect">/pnml/net/page/subnet</dsig-xpath:XPath>
        </ds:Transform>
      </ds:Transforms>
      <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
      <ds:DigestValue>2jmj7l5rSw0yVb/vlWAYkK/YBwk=</ds:DigestValue>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>
    PIwJ414XUFjfd5IYFcJBKeCmYJfanN7Wcus5F5PJRl1yMGVcfeocqQv1nTAn86pQW4NxhYrXEEnD
    zO5Dic/aKC/jt8zgnCZ81DVLCnJLmtc61ltKezEs0ekE6A9PsRjP0DusqtVKL4C2miiFiPsL3enn
    rXBk3ZPYpLcXZw5q/js=
  </ds:SignatureValue>
  <ds:KeyInfo>
    <ds:X509Data>
      <ds:X509Certificate>
        MIICgTCCAeqgAwIBAgIETfh4CTANBgkqhkiG9w0BAQUFADCBhDELMAkGA1UEBhMCRVMxETAPBgNV
        BAgTCExBIFJJT0pBMREwDwYDVQQHDAhMT0dST8KlTzEgMB4GA1UEChMXVU5JVkVSU01EQUQgREUg
        TEEgUklPSkExDDAKBgNVBAsTA1BGQzEfMB0GA1UEAwwWScKlSUdPIExFw6BOIFNBTUFOSUVHTzAe
        Fw0xMTA2MTUw0TE0NDlaFw0xMTA5MTMw0TE0NDlaMIGEMQswCQYDVQQGEwJFUzERMA8GA1UECBMI
        TEEgUklPSkExETAPBgNVBAcMCExPR1JPwqVPMSAwHgYDVQQKExdVTklWRVJTSURBRCBERSBMQSBS
        SU9KQTEMMAoGA1UECxMDUEZDMR8wHQYDVQQDDBZJwqVJRO8gTEXDoE4gU0FNQU5JRUdPMIGfMA0G
        CSqGSIb3DQEBAQUAA4GNADCBiQKBgQChePFNVCIfphFlyXQ9BysiR5BfXIuv3AnAK8OFuw4tTFwC
        nVUjJeGnkUYQO32oUu+fEBK8WsEqjeH8A7zrHTRQjfYZWyuGWrM8gJXOa/POMROPm/c/H8b5a6Nx
        1/+zLwR0tYkqLI2xqDOFII2RwK5L2yGeV4T4y8i3h1U00FTSEwIDAQABMA0GCSqGSIb3DQEBBQUA
        A4GBAID0vAAd0CaTpy+83bGB2KmngMJrNxxWDpAi5LGFrN8iCShmbTpIeIbYBUAaBpZtdhOnhq4n
        wD5QOENSFipQcdH5GEpPM9Rquy6xMwfda9EU5UfOSEmbk4fK2vaIOVjynpQsJ9P99en02smQlyvw
        /hBa7Xacz6qDut8ghUeuV5Js
      </ds:X509Certificate>
    </ds:X509Data>
    <ds:KeyValue>
```

```
        <ds:RSAKeyValue>
          <ds:Modulus>
            oXjxTVQiH6YRZcl0PQcrIkeQX1yLr9wJwCvNBbsOLUxcAp1VIyXhp5FGEDt9qFLvnxASvFrBKo3h
            /A086x00UI32GVsrhlqzPICVzmvz9DETj5v3Px/G+Wujcdf/sy8EdLWJKiyNsagzhSCNkcCuS9sh
            nleE+MvIt4dVNDhU0hM=
          </ds:Modulus>
          <ds:Exponent>AQAB</ds:Exponent>
        </ds:RSAKeyValue>
      </ds:KeyValue>
    </ds:KeyInfo>
  </ds:Signature>
</pnml>
```


5.4 Complete security

In this section I am going to explain several important questions. Until now, I have described the two security operation separately: hiding and signing. I want some parts to be hidden and other parts to be signed. But, what would happen if I want to hide and sign the same parts? In this case, the result is different depending on the order selected. Let's see the differences.

Suppose first cipher and then sign. If this order is applied, the cipher has no problem, but the sign has a obvious but important detail: the signed content is encrypted itself. What does it mean? Well, you don't see what you are signing. The implications are simple: you don't know exactly what you are signing. This order is useful in order to guarantee the integrity, but you have to be sure that what you are signing is exactly what you want.

The other order is: first signing and then ciphering. In this case, signing has no problem. The special characteristic is the ciphering. As explained before in this chapter, after signing, appears a new tag `<Signature>` appended to the original content. Then, if I encrypt the signed content, this tag is still visible but the signature cannot be verified, because the encrypting process replace the original content, so there is a modification of the signed data and it is detected by the signature, avoiding a correct verification of it.

The selected order depends on the responsible of each part. Anyway, there is a W3C recommendation that describes a XMLSignature decryption transform [84] that permits to differentiate between ciphered content that were encrypted before signing (and must not be decrypted) and ciphered content that were encrypted after signing (and must be decrypted). So the XML applications are able to interpret the correct way to validate the signature. As it is a particular case of configuration, and my intention is explain the general guidelines, I am not going to explain it here. Again, the responsible of the net/subnet is who has to decide the concrete configuration, but maintaining the general rules that I explain in this thesis.

5.5 Conclusions

As we have seen, one of the applications of subnetting Petri nets and represent them in PNML is that I can apply standard processes based on XML. In this case, once defined and represented a subnet in PNML, I have applied XMLEncryption in order to hide the internal structure of the subnet and XMLSignature for signing the subnet.

The use of standard and widely extended technologies like XMLEncryption and XMLSignature enables that everybody has access to them, so the processes explained here are perfectly usable.

One important conclusion in the encryption section is that several subnets in the same net can be ciphered with different options. This is used, for example, for one receiver to decrypt the subnet addressed to him but no other one.

Other important use is that with a PNML file with encrypted subnets, I can replace one ciphered subnet by other ciphered subnet without the necessity of decrypt any of them: we can work with encrypted subnets if the interfaces are the same. With the explained in this chapter, the privacy of parts of a Petri net is guarantied.

In XMLSignature, integrity, authentication and non repudiation is assured. A big difference with XMLEncryption is that the signature is attached to the original file, instead of replace part of the net. In this case I have used enveloped signature, but if we use the detached way, the signature is in other file, separating signature and signed content. In signature it makes no sense replace one signature by another, but we can have as many signatures as we want, including different signatures of the same contents.

Using both technologies XMLEncryption and XMLSignature together we can reach very high security levels, but we have to be careful with the order of the processes, because depending on the selected order, the acquired properties will be different.

The last conclusion, but very important one is that the responsible of the net has to choose the concrete configurations of XMLEncryption and XMLSignature, the transforms and the order of the actions in order to achieve his particular objectives.

Chapter 6

Conclusions

Throughout this paper I have enriched Petri nets with definitions and properties. From this initial presentation, have been building a series of elements as a basis for further investigation. We defined subnets, subnets classifications have been studied, we have defined front-ends (interfaces) for those subnets, etc.. From this point is possible a further study of these subnets (their properties, utilities,....) and the methodological study of securing parts of Petri nets.

In the first chapter of this thesis I have introduced the research problem. The nets are represented in a comprehensive way, so that the whole information is visible to everybody. Furthermore, these nets are not prepared to avoid undesired changes or to ensure the authoring of them.

So here is my contribution to the knowledge: to provide security to a Petri net. The aspects covered by this investigation are:

- to occult a part of the net (or entire). The secret is maintained, and all the information is stored in the same file, but hidden. This information is only available to accredited people.
- to avoid unwanted changes. Any modification is, at least, detected.
- to authenticate the net (or a part of it). We know who has developed a Petri net or subnet.
- to avoid the possibility of supplant other people in the authority of the Petri net or some of its parts.

The next chapter is the state of the art. In this chapter, the literature about subnets, hiding, encryption, Petri net representation, PNML extensions, Petri net securing, etc.

are grouped and analyzed in order to understand the general knowledge about these topics, related to my objectives. The general conclusions in this part are:

- There are lots of authors and contents about Petri nets but very few about subnets.
- There is no standard way to represent subnets in a form that is not graphical.
- Many works studied security using Petri nets, but there is not literature about security over Petri nets themselves.
- In particular there is not material on how to hide parts of a Petri nets and either about integrity, authentication or non repudiation.

Once reviewed the state of the art, I enter into the study of subnets. The main goal is to find a structure of Petri subnets that is easily represented in other formats in addition to graphical mode.

In this chapter I explain how to cut a Petri net into several subnets using the incidence matrix. The method of cutting into two subnets is studied. This method split the incidence matrix into four parts: the subnets per se and two other parts that defined the interaction between this two subnets (N_1, N_2, PIM and TIM). This will allow us to define the front-end of a subnet in order to abstract is content from the rest of the net.

Once explained subnets, I make a subnet classification. This classification is based on the structure properties. So we can talk about:

- disjoint subnets, if there are not arcs between the element of the subnet.
- macroplaces, if the only way to enter the subnet is from a transition outside towards a place inside.
- macrotransitions, if the only way to enter the subnet is from a place outside towards a transition inside.
- sinkhole, if there is no arc leaving the subnet.
- source, if there is no arc entering the subnet

Then, one of the main parts of the thesis is studied: the subnet front-end. It is a very important concept because the rest of the thesis is based on it: if I define a subnet with its own front-end, I can know its behaviour without the necessity of knowing the internal structure. In this chapter, I introduce the critical concepts of front-end, input

and output gates from places or transitions and attachable net that are going to be used later.

This part of the Petri subnets theory is finished, so I can go on the representation of this kind of subnets. As one of my goals is to hide parts of a Petri net, but not erasing information, the main problem is to find a way to represent a subnet in order to cipher information. From my point of view, the only alternative nowadays to solve this problem is the use of PNML. Other representations cannot maintain or recover the original information once is hidden.

PNML is a xml standard way to represent Petri nets. But it has a problem: there is noway to represent subnets. So I have explained a possible PNML extension that support this kind of information. The key is the definition of several custom xml tags: `<subnet>`, `<interface>`, `<gate>` and `<content>`. These four tags are enough to my goals. But Petri nets are a really wide knowledge area. So i don't define a closed PNML extension, but a way for each person to extend it as needed.

At this moment I have the basis to secure a Petri net (or parts of it). The next step is the securization itself. The first step is the privacy of the whole net or only of a part. The technology that I have selected to reach this goal is XMLEncryption. As it is an standard, it is widely extended and well known. It has de possibility of symmetric and asymmetric ciphering so the confidentiality of the information is ensured. This method of encryption replace tags in the xml file by encrypted content, only accessible by people that knows the right decrypting key. In this case, the tag replaced is `<content>`, inside the tag `<subnet>`. Once this is done only the front-end of the subnet is exposed, maintaining two properties:

- The structure of the entire net is not affected, because the interaction with elements of the subnet is always through the front-end.
- The information has been not deleted. It is hidden, waiting for somebody with the correct key.
- I can change one subnet by another if they have the same front-end, even though they are ciphered.

The other goals of security are data integrity (not allow unwanted modifications), authentication (know the author or responsible), and non repudiation (nobody can be supplanted by another one). All of these goals can be achieved by a digital signature. In this case I my proposition is the use of XMLSignature, that is going to allow to sign the entire Petri net or only parts of it (the secured ones) by using XPath expressions.

XMLSignature is a standard too, so it has been widely probed and examined by the community. With XMLSignature the signed content is not replaced. Instead of that, a new xml element appears in the PNML file with the necessary information to validate the signature.

And this is the final stage of this thesis. There several ways to follow in further works. For example:

- In this thesis I have worked with very basic Petri nets, with only places, transitions, arcs and weights. This work can be extended to other kind of Petri nets, such as coloured Petri nets, High level Petri nets, fluid Petri nets, ...
- The properties of Petri nets replacing subnets with other subnets can other field of investigation.
- The properties of the subnets have not been studied from the point of view of the behaviour, contemplating markings and evolutions.

Appendix A

PNML grammar

The official PNML grammar is available in www.pnml.org. It is written in RELAX NG format. These are the main parts that I use in this work.

A.1 RELAX NG implementation of PNML Core Model

```
<?xml version="1.0" encoding="UTF-8"?>

<grammar ns="http://www.pnml.org/version-2009/grammar/pnml"
  xmlns="http://relaxng.org/ns/structure/1.0"
  xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
  datatypeLibrary="http://www.w3.org/2001/XMLSchema-datatypes">

  <a:documentation>
    Petri Net Markup Language (PNML) schema.
    RELAX NG implementation of PNML Core Model.

    File name: pnmlcoremodel.rng
    Version: 2009
    (c) 2001-2009
    Michael Weber,
    Ekkart Kindler,
    Christian Stehno,
    Lom Hillah (AFNOR)
    Revision:
    July 2008 - L.H
  </a:documentation>

  <include href="http://www.pnml.org/version-2009/grammar/anyElement.rng"/>

  <start>
    <ref name="pnml.element"/>
  </start>

  <define name="pnml.element">
```

```
  <element name="pnml">
    <a:documentation>
      A PNML document consists of one or more Petri nets.
      It has a version.
    </a:documentation>
    <oneOrMore>
      <ref name="pnml.content"/>
    </oneOrMore>
  </element>
</define>

<define name="pnml.content">
  <ref name="net.element"/>
</define>

<define name="net.element">
  <element name="net">
    <a:documentation>
      A net has a unique identifier (id) and refers to
      its Petri Net Type Definition (PNTD) (type).
    </a:documentation>
    <ref name="identifier.content"/>
    <ref name="nettype.uri"/>
    <a:documentation>
      The sub-elements of a net may occur in any order.
      A net consists of at least a top-level page which
      may contain several objects. A net may have a name,
      other labels (net.labels) and tool specific information in any order.
    </a:documentation>
    <interleave>
      <optional>
        <ref name="Name"/>
      </optional>
      <ref name="net.labels"/>
      <oneOrMore>
        <ref name="page.content"/>
      </oneOrMore>
      <zeroOrMore>
        <ref name="toolspecific.element"/>
      </zeroOrMore>
    </interleave>
  </element>
</define>

<define name="identifier.content">
  <a:documentation>
    Identifier (id) declaration shared by all objects in any PNML model.
  </a:documentation>
  <attribute name="id">
    <data type="ID"/>
  </attribute>
</define>

<define name="nettype.uri">
  <a:documentation>
```

```
      The net type (nettype.uri) of a net should be redefined in the grammar
      for a new Petri net Type.
      An example of such a definition is in ptnet.pntd, the grammar
      for P/T Nets. The following value is a default.
    </a:documentation>
    <attribute name="type">
      <value>http://www.pnml.org/version-2009/grammar/pnmlcoremodel</value>
    </attribute>
  </define>

  <define name="net.labels">
    <a:documentation>
      A net may have unspecified many labels. This pattern should be used
      within a PNTD to define the net labels.
    </a:documentation>
    <empty/>
  </define>

  <define name="basicobject.content">
    <a:documentation>
      Basic contents for any object of a PNML model.
    </a:documentation>
    <interleave>
      <optional>
        <ref name="Name"/>
      </optional>
      <zeroOrMore>
        <ref name="toolspecific.element"/>
      </zeroOrMore>
    </interleave>
  </define>


  <define name="page.content">
    <a:documentation>
      A page has an id. It may have a name and tool specific information.
      It may also have graphical information. It can also have many arbitrary labels.
      Note: according to this definition, a page may contain other pages.
      All these sub-elements may occur in any order.
    </a:documentation>
    <element name="page">
      <ref name="identifier.content"/>
      <interleave>
        <ref name="basicobject.content"/>
        <ref name="page.labels"/>
        <zeroOrMore>
          <ref name="netobject.content"/>
        </zeroOrMore>
        <optional>
          <element name="graphics">
            <ref name="pagegraphics.content"/>
          </element>
        </optional>
      </interleave>
    </element>
```

```
</define>

<define name="netobject.content">
  <a:documentation>
    A net object is either a page, a node or an arc.
    A node is a place or a transition, a reference place of
    a reference transition.
  </a:documentation>
  <choice>
    <ref name="page.content"/>
    <ref name="place.content"/>
    <ref name="transition.content"/>
    <ref name="refplace.content"/>
    <ref name="reftrans.content"/>
    <ref name="arc.content"/>
  </choice>
</define>

<define name="page.labels">
  <a:documentation>
    A page may have unspecified many labels. This pattern should be used
    within a PNTD to define new labels for the page concept.
  </a:documentation>
  <empty/>
</define>

<define name="place.content">
  <a:documentation>
    A place may have several labels (place.labels) and the same content
    as a node.
  </a:documentation>
  <element name="place">
    <ref name="identifier.content"/>
    <interleave>
      <ref name="basicobject.content"/>
      <ref name="place.labels"/>
      <ref name="node.content"/>
    </interleave>
  </element>
</define>

<define name="place.labels">
  <a:documentation>
    A place may have arbitrary many labels. This pattern should be used
    within a PNTD to define the place labels.
  </a:documentation>
  <empty/>
</define>

<define name="transition.content">
  <a:documentation>
    A transition may have several labels (transition.labels) and the same
    content as a node.
  </a:documentation>
  <element name="transition">
```

```
    <ref name="identifier.content"/>
    <interleave>
      <ref name="basicobject.content"/>
      <ref name="transition.labels"/>
      <ref name="node.content"/>
    </interleave>
  </element>
</define>

<define name="transition.labels">
  <a:documentation>
    A transition may have arbitrary many labels. This pattern should be
    used within a PNTD to define the transition labels.
  </a:documentation>
  <empty/>
</define>

<define name="node.content">
  <a:documentation>
    A node may have graphical information.
  </a:documentation>
  <optional>
    <element name="graphics">
      <ref name="nodegraphics.content"/>
    </element>
  </optional>
</define>

<define name="reference">
  <a:documentation>
    Here, we define the attribute ref including its data type.
    Modular PNML will extend this definition in order to change
    the behavior of references to export nodes of module instances.
  </a:documentation>
  <attribute name="ref">
    <data type="IDREF"/>
  </attribute>
</define>

<define name="refplace.content">
  <a:documentation>
    A reference place is a reference node.
  </a:documentation>
  <a:documentation>
    Validating instruction:
    - _ref_ MUST refer to _id_ of a reference place or of a place.
    - _ref_ MUST NOT refer to _id_ of its reference place element.
    - _ref_ MUST NOT refer to a cycle of reference places.
  </a:documentation>
  <element name="referencePlace">
    <ref name="refnode.content"/>
  </element>
</define>

<define name="reftrans.content">
```

```
  <a:documentation>
    A reference transition is a reference node.
  </a:documentation>
  <a:documentation>
    Validating instruction:
    - The reference (ref) MUST refer to a reference transition or to a
    transition.
    - The reference (ref) MUST NOT refer to the identifier (id) of its
    reference transition element.
    - The reference (ref) MUST NOT refer to a cycle of reference transitions.
  </a:documentation>
  <element name="referenceTransition">
    <ref name="refnode.content"/>
  </element>
</define>

<define name="refnode.content">
  <a:documentation>
    A reference node has the same content as a node.
    It adds a reference (ref) to a (reference) node.
  </a:documentation>
  <ref name="identifier.content"/>
  <ref name="reference"/>
  <ref name="basicobject.content"/>
  <ref name="node.content"/>
</define>

<define name="arc.content">
  <a:documentation>
    An arc has a unique identifier (id) and
    refers both to the node's id of its source and
    the node's id of its target.
    In general, if the source attribute refers to a place,
    then the target attribute refers to a transition and vice versa.
  </a:documentation>
  <element name="arc">
    <ref name="identifier.content"/>
    <attribute name="source">
      <data type="IDREF"/>
    </attribute>
    <attribute name="target">
      <data type="IDREF"/>
    </attribute>
    <a:documentation>
      The sub-elements of an arc may occur in any order.
      An arc may have a name, graphical and tool specific information.
      It may also have several labels.
    </a:documentation>
    <interleave>
      <optional>
        <ref name="Name"/>
      </optional>
      <ref name="arc.labels"/>
      <optional>
        <element name="graphics">
```

```
          <ref name="edgegraphics.content"/>
        </element>
      </optional>
      <zeroOrMore>
        <ref name="toolspecific.element"/>
      </zeroOrMore>
    </interleave>
  </element>
</define>

<define name="arc.labels">
  <a:documentation>
    An arc may have arbitrary many labels. This pattern should be used
    within a PNTD to define the arc labels.
  </a:documentation>
  <empty/>
</define>

<define name="pagegraphics.content">
  <a:documentation>
    A page graphics is actually a node graphics
  </a:documentation>
  <ref name="nodegraphics.content"/>
</define>

<define name="nodegraphics.content">
  <a:documentation>
    The sub-elements of a node's graphical part occur in any order.
    At least, there may be one position element.
    Furthermore, there may be a dimension, a fill, and a line element.
  </a:documentation>
  <interleave>
    <ref name="position.element"/>
    <optional>
      <ref name="dimension.element"/>
    </optional>
    <optional>
      <ref name="fill.element"/>
    </optional>
    <optional>
      <ref name="line.element"/>
    </optional>
  </interleave>
</define>

<define name="edgegraphics.content">
  <a:documentation>
    The sub-elements of an arc's graphical part occur in any order.
    There may be zero or more position elements.
    Furthermore, there may be a line element.
  </a:documentation>
  <interleave>
    <zeroOrMore>
      <ref name="position.element"/>
    </zeroOrMore>
```

```
    <optional>
      <ref name="line.element"/>
    </optional>
  </interleave>
</define>

<define name="simpletext.content">
  <a:documentation>
    This definition describes the contents of simple text labels
    without graphics.
  </a:documentation>
  <optional>
    <element name="text">
      <a:documentation>
        A text should have a value.
        If not, then there must be a default.
      </a:documentation>
      <text/>
    </element>
  </optional>
</define>

<define name="annotationstandard.content">
  <a:documentation>
    The definition annotationstandard.content describes the
    standard contents of an annotation.
    Each annotation may have graphical or tool specific information.
  </a:documentation>
  <interleave>
    <optional>
      <element name="graphics">
        <ref name="annotationgraphics.content"/>
      </element>
    </optional>
    <zeroOrMore>
      <ref name="toolspecific.element"/>
    </zeroOrMore>
  </interleave>
</define>

<define name="simpletextlabel.content">
  <a:documentation>
    A simple text label is an annotation to a net object containing
    arbitrary text.
    Its sub-elements occur in any order.
    A simple text label behaves like an attribute to a net object.
    Furthermore, it contains the standard annotation contents which
    basically defines the graphics of the text.
  </a:documentation>
  <interleave>
    <ref name="simpletext.content"/>
    <ref name="annotationstandard.content"/>
  </interleave>
</define>
```

```
<define name="Name">
  <a:documentation>
    Label definition for a user given name of an
    element.
  </a:documentation>
  <element name="name">
    <ref name="simpletextlabel.content"/>
  </element>
</define>

<define name="annotationgraphics.content">
  <a:documentation>
    An annotation's graphics part requires an offset element describing
    the offset the center point of the surrounding text box has to
    the reference point of the net object on which the annotation occurs.
    Furthermore, an annotation's graphic element may have a fill, a line,
    and font element.
  </a:documentation>
  <ref name="offset.element"/>
  <interleave>
    <optional>
      <ref name="fill.element"/>
    </optional>
    <optional>
      <ref name="line.element"/>
    </optional>
    <optional>
      <ref name="font.element"/>
    </optional>
  </interleave>
</define>

<define name="position.element">
  <a:documentation>
    A position element describes Cartesian coordinates.
  </a:documentation>
  <element name="position">
    <ref name="coordinate.attributes"/>
  </element>
</define>

<define name="offset.element">
  <a:documentation>
    An offset element describes Cartesian coordinates.
  </a:documentation>
  <element name="offset">
    <ref name="coordinate.attributes"/>
  </element>
</define>

<define name="coordinate.attributes">
  <a:documentation>
    The coordinates are decimal numbers and refer to an appropriate
    xy-system where the x-axis runs from left to right and the y-axis
    from top to bottom.
```

```
    </a:documentation>
    <attribute name="x">
      <data type="decimal"/>
    </attribute>
    <attribute name="y">
      <data type="decimal"/>
    </attribute>
  </define>

  <define name="dimension.element">
    <a:documentation>
      A dimension element describes the width (x coordinate) and height
      (y coordinate) of a node.
      The coordinates are actually positive decimals.
    </a:documentation>
    <element name="dimension">
      <attribute name="x">
        <ref name="positiveDecimal.content"/>
      </attribute>
      <attribute name="y">
        <ref name="positiveDecimal.content"/>
      </attribute>
    </element>
  </define>

  <define name="positiveDecimal.content" ns="http://www.w3.org/2001/XMLSchema-datatypes">
    <a:documentation>
      Definition of a restricted positive decimals domain with a total digits
      number of 4 and 1 fraction digit. Ranges from 0 to 999.9
    </a:documentation>
    <data type='decimal'>
      <param name='totalDigits'>4</param>
      <param name='fractionDigits'>1</param>
      <param name='minExclusive'>0</param>
    </data>
  </define>

  <define name="fill.element">
    <a:documentation>
      A fill element describes the interior colour, the gradient colour,
      and the gradient rotation between the colors of an object.  If an
      image is available the other attributes are ignored.
    </a:documentation>
    <element name="fill">
      <optional>
        <attribute name="color">
          <ref name="color.type"/>
        </attribute>
      </optional>
      <optional>
        <attribute name="gradient-color">
          <ref name="color.type"/>
        </attribute>
      </optional>
      <optional>
```

```
      <attribute name="gradient-rotation">
        <choice>
          <value>vertical</value>
          <value>horizontal</value>
          <value>diagonal</value>
        </choice>
      </attribute>
    </optional>
    <optional>
      <attribute name="image">
        <data type="anyURI"/>
      </attribute>
    </optional>
  </element>
</define>

<define name="line.element">
  <a:documentation>
    A line element describes the shape, the colour, the width, and the
    style of an object.
  </a:documentation>
  <element name="line">
    <optional>
      <attribute name="shape">
        <choice>
          <value>line</value>
          <value>curve</value>
        </choice>
      </attribute>
    </optional>
    <optional>
      <attribute name="color">
        <ref name="color.type"/>
      </attribute>
    </optional>
    <optional>
      <attribute name="width">
        <ref name="positiveDecimal.content"/>
      </attribute>
    </optional>
    <optional>
      <attribute name="style">
        <choice>
          <value>solid</value>
          <value>dash</value>
          <value>dot</value>
        </choice>
      </attribute>
    </optional>
  </element>
</define>

<define name="color.type">
  <a:documentation>
    This describes the type of a color attribute. Actually, this comes
```

```
    from the CSS2 (and latest versions) data type system.
  </a:documentation>
  <text/>
</define>

<define name="font.element">
  <a:documentation>
    A font element describes several font attributes, the decoration,
    the alignment, and the rotation angle of an annotation's text.
    The font attributes (family, style, weight, size) should be conform
    to the CSS2 and latest versions data type system.
  </a:documentation>
  <element name="font">
    <optional>
      <attribute name="family">
        <text/>  <!-- actually, CSS2 and latest versions font-family -->
      </attribute>
    </optional>
    <optional>
      <attribute name="style">
        <text/>  <!-- actually, CSS2 and latest versions font-style -->
      </attribute>
    </optional>
    <optional>
      <attribute name="weight">
        <text/>  <!-- actually, CSS2 and latest versions font-weight -->
      </attribute>
    </optional>
    <optional>
      <attribute name="size">
        <text/>  <!-- actually, CSS2 and latest versions font-size -->
      </attribute>
    </optional>
    <optional>
      <attribute name="decoration">
        <choice>
          <value>underline</value>
          <value>overline</value>
          <value>line-through</value>
        </choice>
      </attribute>
    </optional>
    <optional>
      <attribute name="align">
        <choice>
          <value>left</value>
          <value>center</value>
          <value>right</value>
        </choice>
      </attribute>
    </optional>
    <optional>
      <attribute name="rotation">
        <data type="decimal"/>
      </attribute>
```

```
      </optional>
    </element>
  </define>

  <define name="toolspecific.element">
    <a:documentation>
      The tool specific information refers to a tool and its version.
      The further substructure is up to the tool.
    </a:documentation>
    <element name="toolspecific">
      <attribute name="tool">
        <text/>
      </attribute>
      <attribute name="version">
        <text/>
      </attribute>
      <zeroOrMore>
        <ref name="anyElement"/>
      </zeroOrMore>
    </element>
  </define>

</grammar>
```

Listing A.1: RELAX NG implementation of PNML Core Model

A.2 RELAX NG implementation of Petri Net Type Definition for Place/Transition nets

```
<?xml version="1.0" encoding="UTF-8"?>

<grammar ns="http://www.pnml.org/version-2009/grammar/pnml"
  xmlns="http://relaxng.org/ns/structure/1.0"
  xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0">

  <a:documentation>
    RELAX NG implementation of Petri Net Type Definition for Place/Transition nets.
    This PNTD re-defines the value of nettype.uri for P/T nets.

    File name: ptnet.pntd
    Version: 2009
    (c) 2007-2009
    Lom Hillah (AFNOR)
    Revision:
    July 2008 - L.H
  </a:documentation>

  <a:documentation>
    The PT Net type definition.
    This document also declares its namespace.
    All labels of this Petri net type come from the Conventions document.
    The use of token graphics as tool specific feature is possible.
  </a:documentation>

  <include href="http://www.pnml.org/version-2009/grammar/conventions.rng"/>

  <!--
  <include href="http://www.pnml.org/version-2009/grammar/pnmlextensions.rng"/>
    We do not need to include this, because the pnmlcoremodel.rng covers any
    toolspecific extension.
  -->

  <include href="http://www.pnml.org/version-2009/grammar/pnmlcoremodel.rng"/>

  <define name="nettype.uri" combine="choice">
    <a:documentation>
      The URI value for the net type attribute,
      declaring the type of P/T nets.
    </a:documentation>
    <attribute name="type">
      <value>http://www.pnml.org/version-2009/grammar/ptnet</value>
    </attribute>
  </define>


  <define name="PTMarking">
    <a:documentation>
      Label definition for initial marking in nets like P/T-nets.
      <contributed>Michael Weber</contributed>
      <date>2003-06-16</date>
```

```
      <reference>
        W. Reisig: Place/transition systems. In: LNCS 254. 1987.
      </reference>
    </a:documentation>
    <element name="initialMarking">
      <ref name="nonnegativeintegerlabel.content"/>
    </element>
  </define>

  <define name="PTArcAnnotation">
    <a:documentation>
      Label definition for arc inscriptions in P/T-nets.
      <contributed>Michael Weber, AFNOR</contributed>
      <date>2003-06-16</date>
      <reference>
        W. Reisig: Place/transition systems. In: LNCS 254. 1987.
      </reference>
    </a:documentation>
    <element name="inscription">
      <ref name="positiveintegerlabel.content"/>
    </element>
  </define>

  <define name="place.labels" combine="interleave">
    <a:documentation>
      A place of a P/T net may have an initial marking.
    </a:documentation>
    <optional><ref name="PTMarking"/></optional>
  </define>

  <define name="arc.labels" combine="interleave">
    <a:documentation>
      An arc of a P/T net may have an inscription.
    </a:documentation>
    <optional><ref name="PTArcAnnotation"/></optional>
  </define>

</grammar>
```

Listing A.2: RELAX NG implementation of PNTD for Place/Transition nets

Bibliography

[1] E. Jiménez Macías and M. Pérez de la Parte. Simulation and optimization of logistic and production systems using discrete and continuous petri nets. Simulation, 80 (3):143–152, 2004.

[2] T. Guasch, M A. Piera, J. Casanovas, and J. Figueras. Modelado y Simulación. Aplicación a procesos logísticos de fabricación y servicios. Edicions UPC, Barcelona, Spain, 2002.

[3] K Jensen and L.M Kristensen. Coloured Petri Nets: Modelling and validation of concurrent systems. 2009. doi: 10.1007/b95112.

[4] I. León Samaniego. Seguridad y protección en envío y almacenamiento de datos. firmado y cifrado. aplicación a redes de petri y gestión de residuos con e3l. Master's thesis, Universidad de La Rioja. Logroño, Spain, 2011.

[5] R. Valette. Analysis of petri nets by stepwise refinements. Journal of Computer and System Sciences, 18(1):35–46, 1979. doi: 10.1016/0022-0000(79)90050-3.

[6] I Suzuki and T Murata. A method for stepwise refinement and abstraction of petri nets. Journal of Computer and System Sciences, 27(1):51–76, 1983. doi: 10.1016/0022-0000(83)90029-6.

[7] H.M.A. Fahmy. Analysis of petri nets by partitioning: Splitting transitions. Theoretical Computer Science, 77(3):321–330, 1990. doi: 10.1016/0304-3975(90) 90174-G.

[8] Sa Druzhinin, Va; Yuditskii. Construction of well-formed petri nets from standard subnets. Automation and Remote Control, 53(12):1922–1927, 1992.

[9] Hossam Mahmoud Ahmad Fahmy. Analysis of petri nets by partitioning: splitting places or transitions. International Journal of Computer Mathematics, 48(3-4): 127–148, 1993.

[10] C Xia. Analysis and application of petri subnet reduction. Procs. of the IEEE, 6 (8):1662–1669, 2011.

[11] Tadao Murata. Petri nets: properties, analysis and applications. Proceedings of the IEEE, 77(4):541–580, 1989. doi: 10.1109/5.24143.

[12] M Silva. Las Redes de Petri: en la Automática y en la Informática. Ed. AC, Madrid, Spain, 1985.

[13] M Silva. In Practice of Petri Nets in Manufacturing. Chapman and Hall, London, UK, 1993.

[14] R. David and H. Alla. Discrete, Continuous and Hybrid Petri Nets. Springer, Berlin, Germany, 1st ed., 2004 edition, 2010.

[15] JL Peterson. Petri Net Theory and the Modeling of Systems. Prentice Hall, Englewood Clifs, NJ, 1981.

[16] Carl Adam Petri. Kommunication mit Automaten. PhD thesis, Technischen Hoschule Darmstadt, 1962.

[17] Carl Adam Petri. Communication with automata. Technical Report 65-377, Rome Air Devolopement Center, 1966.

[18] Carl Adam Petri. Interpretations of net theory. Technical Report 75-07, Gessellschaft fur Mathematik und Datenverarbeitung, Bonn, 1976.

[19] Carl Adam Petri and E; Smith. The pragmatic dimension of net theory. In In procs. of the 8th European workshop on Applications and Theory of Petri Nets, 2007.

[20] E. Jiménez, J. Júlyez, L. Recalde, and M. Silva. Relaxed continuous views of discrete event systems: Considerations on forrester diagrams and petri nets. volume 5, pages 4897–4904, 2004. doi: 10.1109/ICSMC.2004.1401307.

[21] L.E. Holloway, B.H. Krogh, and A. Giua. A survey of petri net methods for controlled discrete event systems. Discrete Event Dynamic Systems: Theory and Applications, 7(2):151–190, 1997.

[22] J.I. Latorre, E. Jiménez, M. Pérez, J. Blanco, and E. Martínez. The alternatives aggregation petri nets as a formalism to design discrete event systems. International Journal of Simulation and Process Modelling, 6(2):152–164, 2010. doi: 10.1504/IJSPM.2010.036019.

[23] J.I. Latorre, E. Jiménez, and M. Pérez. Coloured petri nets as a formalism to represent alternative models for a discrete event system. pages 247–252, 2010.

[24] M. Silva, J. Júlvez, C. Mahulea, and C.R. Vázquez. On fluidization of discrete event models: Observation and control of continuous petri nets. Discrete Event

Dynamic Systems: Theory and Applications, 21(4):427–497, 2011. doi: 10.1007/s10626-011-0116-9.

[25] L. Recalde, E. Teruel, and M. Silva. Modeling and analysis of sequential processes that cooperate through buffers. IEEE Transactions on Robotics and Automation, 14(2):267–277, 1998. doi: 10.1109/70.681245.

[26] K. Jensen, L.M. Kristensen, and L. Wells. Coloured petri nets and cpn tools for modelling and validation of concurrent systems. International Journal on Software Tools for Technology Transfer, 9(3-4):213–254, 2007. doi: 10.1007/s10009-007-0038-x.

[27] L.M. Kristensen and K. Jensen. Teaching modelling and validation of concurrent systems using coloured petri nets. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 5100 LNCS:19–34, 2008. doi: 10.1007/978-3-540-89287-8-2.

[28] R Silva, M; Valette. Petri nets and flexible manufacturing. Advances in Petri Nets, 424:374–417, 1989.

[29] A. Desrochers and R.Y. Al-Jaar. Applications of Petri Nets in Manufacturing Systems. Modeling, Control ad Performance Analysis. IEEE Press, New York, USA, 2010.

[30] M. Silva and E. Teruel. Petri nets for the design and operation of manufacturing systems. European Journal of Control, 3(3):182–199, 1997.

[31] M. Silva. 50 years after the phd thesis of carl adam petri: A perspective. pages 13–20, 2012.

[32] G. Balbo and M. Silva. Performance Models for Discrete Event Systems with Synchronizations: Formalisms and Analysis Techniques. Editorial Kronos, Zaragoza, Spain, 1998.

[33] E. Jiménez, M. Pérez, and I. Latorre. Industrial applications of petri nets: System modelling and simulation. pages 159–164, 2006.

[34] J.I. Latorre, E. Jiménez, and M. Pérez. The optimization problem based on alternatives aggregation petri nets as models for industrial discrete event systems. Simulation, 89(3):346–361, 2013. doi: 10.1177/0037549712464410.

[35] Tadao Murata. State equation, controllability, and maximal matchings of petri nets. IEEE Transactions on Automatic Control, AC-22(3):412–416, 1977.

[36] Tadao Murata. Petri nets, marked graphs, and circuit-system theory. Circuits Syst, 11(3):2–12, 1977.

[37] Manuel Silva. Introducing Petri nets. Chapman and Hall, 1993.

[38] Y. Lien. Termination properties of generalized petri nets. SIAM J. Comput. 5, 5 (2):251–265, 1976.

[39] Kurt Jensen. Introduction to high-level petri nets. pages 723–726, 1985.

[40] M. Silva and L. Recalde. Petri nets and integrality relaxations: A view of continuous petri net models. IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, 32(4):314–327, 2002. doi: 10.1109/TSMCC.2002.806063.

[41] V. Khomenko and M. Koutny. Branching processes of high-level petri nets. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2619:458–472, 2003.

[42] A.V. Ratzer, L. Wells, H.M. Lassen, M. Laursen, J.F. Qvortrup, M.S. Stissing, M. Westergaard, S. Christensen, and K. Jensen. Cpn tools for editing, simulating, and analysing coloured petri nets. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2679:450–462, 2003.

[43] L.M. Kristensen, J.B. Jorgensen, and K. Jensen. Application of coloured petri nets in system development. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 3098: 626–685, 2004.

[44] M. Silva and L. Recalde. On fluidification of petri nets: From discrete to hybrid and continuous models. Annual Reviews in Control, 28(2):253–266, 2004. doi: 10.1016/j.arcontrol.2004.05.002.

[45] J. Campos and M. Silva. Structural techniques and performance bounds of stochastic petri net models. Advances in Petri Nets, 609:352–391, 1992.

[46] L Recalde, S. Haddad, and M. Silva. Continuous petri nets: Expressive power and decidability issues. International Journal of Foundations of Computer Science, 21 (2):235–256, 2010. doi: 10.1142/S0129054110007222.

[47] E. Fraca, J. Júlvez, and M. Silva. Marking homothetic monotonicity and fluidization of untimed petri nets. pages 21–27, 2012.

[48] C.R. Vázquez and M. Silva. Stochastic continuous petri nets: An approximation of markovian net models. IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans, 42(3):641–653, 2012. doi: 10.1109/TSMCA.2011.2172416.

[49] C.R. Vázquez, C. Mahulea, J. Júlvez, and M. Silva. Introduction to fluid petri nets. Lecture Notes in Control and Information Sciences, 433:365–386, 2013. doi: 10.1007/978-1-4471-4276-8-18.

[50] Tadao Murata. State equation, controllability, and maximal matchings of petri nets. IEEE Transactions on Automatic Control, AC-22(3):412–416, 1977.

[51] J. Engelfriet. Branching processes of petri nets. Acta Informatica, 28(6):575–591, 1991. doi: 10.1007/BF01463946.

[52] M. Silva and T. Murata. B-fairness and structural b-fairness in petri net models of concurrent systems. Journal of Computer and System Sciences, 44(3):447–477, 1992. doi: 10.1016/0022-0000(92)90013-9.

[53] L. Recalde, E. Teruel, and M. Silva. On linear algebraic techniques for liveness analysis of p/t systems. Journal of Circuits, Systems and Computers, 8(1):223–265, 1998.

[54] Q.-T. Zeng and Z.-H. Wu. Process net system of petri net. Jisuanji Xuebao/Chinese Journal of Computers, 25(12):1308–1315, 2002.

[55] E. Teruel and M. Silva. Structure theory of equal conflict systems. Theoretical Computer Science, 153(1-2):271–300, 1996.

[56] F.-S. Hsieh. Robustness analysis of non-ordinary petri nets for flexible assembly/disassembly processes based on structural decomposition. International Journal of Control, 84(3):496–510, 2011. doi: 10.1080/00207179.2011.561443.

[57] G.S. Hura. State space representation of petri nets. Microelectronics Reliability, 24(5):865–868, 1984. doi: 10.1016/0026-2714(84)90009-X.

[58] N.A. Anisimov and V.L. Perchuk. Representation of exchange protocols and petri using finite sequential machine nets. Soviet journal of computer and systems sciences, 24(3):90–95, 1986.

[59] Sajal Das, V.K. Agrawal, Dilip Sarkar, L.M. Patnaik, and P.S. Goel. Reflexive incidence matrix (rim) representation of petri nets. IEEE Transactions on Software Engineering, SE-13(6):643–653, 1987.

[60] V.D. Malyugin. Arithmethical representation of petri nets. Automation and Remote Control, 48(5):696–703, 1987.

[61] R.P. Kaushal, N. Chammas, and H. Singh. A new formulation for state equation representation for petri nets. Microelectronics Reliability, 32(8):1083–1090, 1992. doi: 10.1016/0026-2714(92)90029-K.

[62] D. Kiritsis and P. Xirouchakis. A matrix implementation of petri nets for process planning. pages 173–179, 2001.

[63] Pnml.org - pnml reference site, 2009. URL http://www.pnml.org/. [Online; accessed 21-May-2015].

[64] Iso/iec 15909-1:2004 - systems and software engineering – high-level petri nets – part 1: Concepts, definitions and graphical notation, 2004. URL http://www.iso.org/iso/catalogue_detail.htm?csnumber=43538. [Online; accessed 21-May-2015].

[65] Lom Hillah, Fabrice Kordon, Laure Petrucci-Dauchy, and Nicolas Trèves. PN standardisation: A survey. In Formal Techniques for Networked and Distributed Systems - FORTE 2006, 26th IFIP WG 6.1 International Conference, Paris, France, September 26-29, 2006., pages 307–322, 2006. doi: 10.1007/11888116_23. URL http://dx.doi.org/10.1007/11888116_23.

[66] J. Billington, S. Christensen, K. Van Hee, E. Kindler, O. Kummer, L. Petrucci, R. Post, C. Stehno, and M. Weber. The petri net markup language: Concepts, technology, and tools. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2679: 483–505, 2003.

[67] Iso/iec 15909-2:2011 - systems and software engineering – high-level petri nets – part 2: Transfer format, 2011. URL http://www.iso.org/iso/catalogue_detail.htm?csnumber=43538. [Online; accessed 21-May-2015].

[68] E. Kindler. The epnk: An extensible petri net tool for pnml. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 6709 LNCS:318–327, 2011. doi: 10.1007/978-3-642-21834-7_18.

[69] L.-M. Hillah, F. Kordon, C. Lakos, and L. Petrucci. Extending pnml scope: A framework to combine petri nets types. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7400 LNCS:46–70, 2012. doi: 10.1007/978-3-642-35179-2_3.

[70] F. Moutinho, L. Gomes, F. Ramalho, J. Figueiredo, J.P. Barros, P. Barbosa, R. Pais, and A. Costa. Ecore representation for extending pnml for input-output place-transition nets. pages 2156–2161, 2010. doi: 10.1109/IECON.2010.5675332.

[71] J. Ribeiro, F. Moutinho, F. Pereira, J.P. Barros, and L. Gomes. An ecore based petri net type definition for pnml iopt models. pages 777–782, 2011. doi: 10.1109/INDIN.2011.6034992.

[72] C. Mahulea, J. Júlvez, C.R. Vázquez, and M. Silva. Continuous petri nets: Observability and diagnosis. Lecture Notes in Control and Information Sciences, 433: 387–406, 2013. doi: 10.1007/978-1-4471-4276-8-19.

[73] A. Saboori and C.N. Hadjicostis. Opacity verification in stochastic discrete event systems. pages 6759–6764, 2010. doi: 10.1109/CDC.2010.5717580.

[74] S. Velilla and M. Silva. The spy: A mechanism for safe implementation of highly concurrent systems. Annual Review in Automatic Programming, 14(PART 1):75–81, 1988. doi: 10.1016/0066-4138(88)90012-2.

[75] Xml encryption syntax and processing version 1.1, 2013. URL http://www.w3.org/TR/xmlenc-core1/. [Online; accessed 20-Jun-2015].

[76] Xml signature syntax and processing version 1.1, 2013. URL http://www.w3.org/TR/xmldsig-core1/. [Online; accessed 20-Jun-2015].

[77] J.I. Latorre. An integrated methodology to state and solve optimization problems with Petri nets as disjunctive constraints for decision-making support. PhD thesis, University of La Rioja, 2011.

[78] Extensible markup language (xml) 1.1, 2004. URL http://www.w3.org/TR/2004/REC-xml11-20040204/#dtd. [Online; accessed 15-Jun-2015].

[79] Xml schema part 1: Structures second edition, 2004. URL http://www.w3.org/TR/xmlschema-1/. [Online; accessed 15-Jun-2015].

[80] Xml schema part 2: Datatypes second edition, 2004. URL http://www.w3.org/TR/xmlschema-1/. [Online; accessed 15-Jun-2015].

[81] Relax ng home page, 2014. URL http://relaxng.org/. [Online; accessed 15-Jun-2015].

[82] Xml path language (xpath), 1999. URL http://www.w3.org/TR/xpath/. [Online; accessed 20-Jun-2015].

[83] Xml-signature xpath filter 2.0, 2002. URL http://www.w3.org/TR/xmldsig-filter2/. [Online; accessed 20-Jun-2015].

[84] Decryption transform for xml signature, 2002. URL http://www.w3.org/TR/xmlenc-decrypt. [Online; accessed 25-Jun-2015].