

# Report Templates

7.12.2010

## 1 Introduction

Report templates generate RTF reports using a template written in RTF, and merging the text with information provided by PILAR.

The template processor copies most of the text from the template into the report without modification. However, there may be placeholders to call PILAR for data.

The placeholders have the following syntax

```
<pilar> command(s) </pilar>
```

The format is XML, with a number of tags, to describe the action, and zero or more xml attributes.

```
attr
```

means

```
attr="value"
```

When the attribute is optional, it is presented within brackets

```
[ attr ]
```

and it means that it may be omitted in the placeholder, and PILAR will use a default value.

Most frequently, the attribute value comes between double quotes; but sometimes word is not very friendly with those characters, and tries to replace with left- and right- double quotes. In those cases, you may use whatever character you wish as attribute delimiter. For instance

```
attr=$value$
```

```
attr=%value%
```

```
attr=:value:
```

Please, avoid the following characters that are heavily used by RTF to encode formatting:

```
{ } \
```

and these ones, open and close quotes, that use to have special typographic variants:

```
“ ” „ ‘ ’ `
```

## 2 Text substitution

```
<define kw > value </define>
```

It associates the "kw" with the "value". In the <replace> command, the "kw" will be replaced.

Example: <pilar><define kw="WS">William Shakespeare</define></pilar>

```
<replace kw />
```

Prints the value associated to the "kw".

Example: `<pilar><replace kw="WS" /></pilar>`

`<date [ format ] />`

Prints current date.

If no format is specified, the default is "day.month.year".

You may specify a format as in

<http://java.sun.com/javase/6/docs/api/java/util/Formatter.html>

PILAR will pass a single argument that is the current date.

pattern	output
<code>&lt;pilar&gt;&lt;date /&gt;&lt;/pilar&gt;</code>	4.12.2009
<code>&lt;pilar&gt;&lt;date format="%tc"/&gt;&lt;/pilar&gt;</code>	Fri Dec 04 16:34:01 CET 2009
<code>&lt;pilar&gt;&lt;date format="%tD"/&gt;&lt;/pilar&gt;</code>	12/04/09
<code>&lt;pilar&gt;&lt;date format="%tY/%&lt;tm/%&lt;td"/&gt;&lt;/pilar&gt;</code>	2009/12/04

## 2.1 Predefined text

library	[code] name
library.code	code
library.name	name
model	[code] name
model.code	code
model.name	name
model.description	description
model.kw.KEY	value assigned to KEY

## 3 Library information

`<dimensions.list />`

Prints the list of dimensions that are not OFF.

`<maturity.list />`

Prints the list of maturity levels.

## 4 Assets

`<assets.group name [ asset ] [ family ] [ layer ] [ domain ] />`

attribute	contents	default
<b>name</b>	name of the group	n.a.
<b>asset</b>	list of asset codes, or names of groups of assets	all the assets
<b>family</b>	list of class codes	all the classes
<b>layer</b>	list of layer codes	all the layers
<b>domain</b>	list of security domain codes	all the security domains

Defines a group of assets, and it may be later identified by the given name.

The attributes filter the assets that match one of the families (if any) and one of the layers (if any), and one of the domains (if any).

Examples:

- family="HW"  
all the assets of class HW
- family="HW" domain="base"  
all the assets of class HW in the base domain
- asset="SVR\_01, SVR\_02"  
those two assets

`<assets.list [ asset ] [ family ] [ layer ] [ domain ] />`

attribute	contents	default
<b>asset</b>	list of asset codes, or names of groups of assets	all the assets
<b>family</b>	list of class codes	all the classes
<b>layer</b>	list of layer codes	all the layers
<b>domain</b>	list of security domain codes	all the security domains

Prints the selected assets. It prints one line per asset: [code] name.

The attributes filter the assets that match one of the families (if any) and one of the layers (if any), and one of the domains (if any).

`<assets.description [ asset ] [ family ] [ layer ] [ domain ] />`

attribute	contents	default
<b>asset</b>	list of asset codes, or names of groups of assets	all the assets
<b>family</b>	list of class codes	all the classes
<b>layer</b>	list of layer codes	all the layers

<b>domain</b>	list of security domain codes	all the security domains
---------------	-------------------------------	--------------------------

Prints the description of the selected assets. It prints a few paragraphs, with all the identification of the asset.

The attributes filter the assets that match one of the families (if any) and one of the layers (if any), and one of the domains (if any).

`<assets.valuation [ asset ] [ family ] [ layer ] [ domain ] />`

<b>attribute</b>	<b>contents</b>	<b>default</b>
<b>asset</b>	list of asset codes, or names of groups of assets	all the assets
<b>family</b>	list of class codes	all the classes
<b>layer</b>	list of layer codes	all the layers
<b>domain</b>	list of security domain codes	all the security domains

Prints the valuation of the selected assets. It prints the value and the valuation criteria.

The attributes filter the assets that match one of the families (if any) and one of the layers (if any), and one of the domains (if any).

## 5 Security domains

`<domains.list />`

Lists the security domains in the model.

`<domains.description [ domain ] />`

Lists the description of the security domains.

`<domains.valuation [ domain ] />`

Lists the valuation of the domains.

`<domains.vulnerability [ domain ] />`

List the vulnerabilities identified per domain.

## 6 Interruption steps

`<steps.list />`

List of interruption steps.

## 7 Category

The category of an information system is defined as the highest valuation of its assets in any dimension of security.

`<category [ domain ] />`

The following table shows the algorithm:

category	if ...
<b>HIGH</b>	if valuation of some asset in the domain is higher than [5]
<b>MEDIUM</b>	if valuation of some asset in the domain is higher than [2]
<b>LOW</b>	if valuation of some asset in the domain is higher than [0]

## 8 Threats

`<threats.group name [ threat ] />`

attribute	contents	default
<b>name</b>	name of the group	n.a.
<b>threat</b>	list of threat codes	all the threats that are not OFF

Defines a group of threats, and it may be later identified by the given name.

"threat" may specify a list of threats, by their codes, separated by commas.

`<threats.domain [ domain ] [ dimension ] [ threat ] [ freq ] [ deg ] />`

attribute	contents	default
<b>domain</b>	list of security domain codes	all the security domains
<b>dimension</b>	list of security dimension codes	all the dimensions that are not OFF
<b>threat</b>	list of threat codes	all the threats that are not OFF
<b>freq</b>	true to list likelihood	true
<b>deg</b>	true to list degradation	true

Dumps a compact table per domain.

There is a row per threat specified (either by threat code, or by threat group name).

If "freq" is true, there is one column showing the likelihood of the threat.

If "deg" is true, there is one column per dimension showing the degradation caused by the threat.

There is one column per dimension showing the risk on the domain caused by this threat.

## 9 Safeguards

`<safeguards.group name [ safeguard ] />`

Defines a group of safeguards, and it may be later identified by the given name.  
 "safeguard" may specify a list of safeguards, by their codes, separated by commas.

**< safeguards.list [ safeguard ] [ domain ] [ depth ] [ expertise ] />**

Lists safeguards that apply to one or more domains, down to a certain depth in the tree, or to a given expertise expansion.

"domain" may specify one or more domains to list. If none specified, PILAR lists all of them.

"depth" specifies a depth of expansion. If there is no "depth", then it expands to the specified "expertise". If there is no expertise, it expands to the model expertise.

Expertise is introduced as a number

0 (for basic), 1 (for medium), 2 (for expert).

**< safeguards.required [ safeguard ] [ domain ] [ depth ] [ expertise ] />**

Generates a table showing whether the safeguard applies or not in the domain.  
 See < safeguards.list >.

**< safeguards.valuation [ safeguard ] [ domain ] [ depth ] [ expertise ] [ phase ] />**

Generates a table with the valuation of the safeguards in the specified phases.  
 See < safeguards.list >.

## 9.1 Attributes

attribute	default	examples
<b>name</b>	n.a.	"physical"
<b>safeguard</b>	all	safeguard="SW,HW"
<b>exclude</b>	nothing	exclude="P"
<b>domain</b>	all	domain="logical_security"
<b>depth</b>	full	depth="3"
<b>expertise</b>	model expertise	expertise="0" expertise="basic"
<b>phase</b>	all	phase="current, target"

## 10 Policies

Policies are a subset of safeguards. Therefore, patterns for policies are parallel to the patterns for safeguards.

**< policies.list [ safeguard ] [ domain ] />**

Prints the safeguards related to policy in one or more security domains.

`<policies.required [ safeguard ] [ domain ] />`

Prints a table for the policy safeguards stating whether it applies or not in the domain.

`<policies.valuation [ safeguard ] [ domain ] [ phase ] />`

Prints a table for the policy safeguards stating the maturity of the implementation for the domain and phase.

## 10.1 Attributes

attribute	default	example
safeguard	all	safeguard="SW,HW"
domain	all	domain="logical_security"
phase	all	phase="current, target"

## 11 Procedures

Procedures are a subset of safeguards. Therefore, patterns for procedures are parallel to the patterns for safeguards.

`<procedures.list [ safeguard ] [ domain ] />`

Prints the safeguards related to procedures in one or more security domains.

`<procedures.required [ safeguard ] [ domain ] />`

Prints a table for the procedure safeguards stating whether it applies or not in the domain.

`<procedures.valuation [ safeguard ] [ domain ] [ phase ] />`

Prints a table for the procedure safeguards stating the maturity of the implementation for the domain and phase.

## 11.1 Attributos

attribute	default	example
safeguard	all	safeguard="SW,HW"
domain	all	domain="logical_security"
phase	all	phase="current, target"

## 12 Risk

`<risk.domain [ domain ] [ dimension ] [ phase ] />`

attribute	default	examples
<b>domain</b>	all	domain="logical_security"
<b>dimension</b>	all	dimension="I, C"
<b>phase</b>	all	phase="current, target"

Inserts a table showing the risk in the requested domains, in the requested dimensions, in the requested phases.

Use phase "null" for potential risk.

**<risk.accumulated [ asset ] [ phase1 ] />**

attribute	default	examples
<b>asset</b>	all	asset="HW"
<b>phase1</b>	null	phase1="current"

Inserts a table showing the accumulated risk in the selected phases.

Use phase "null" for potential risk.

**<risk.deflected [ phase1 ] />**

attribute	default	examples
<b>phase1</b>	null	phase1="current"

Inserts a table showing the deflected risk in the selected phases.

Use phase "null" for potential risk.

## 13 Security profiles (EVL)

**<evl.group code name [ control ] />**

Groups some controls under a single name.

**<evl.list code [ control ] [ domain ] [ depth ]  
[ questions ] [ safeguards ] [ applies ] />**

Inserts a list with the controls that apply from the evaluation profile "code", in the requested domains.

"depth" determines how deep we dig.

"questions" (boolean) to specify whether questions are to be listed or not.

"safeguards" (boolean) to specify whether safeguards are to be listed or not.

```
<evl.required code [ control ] [ domain ] [ depth ] [ mode ]
[ questions ] [ safeguards ] [ applies ] />
```

Inserts a table with the applicability of the controls.

“depth” determines how deep we dig.

```
<evl.valuation code [ control ] [ domain ] [ depth ] [ phase ] [ mode ]
[ questions ] [ safeguards ] [ applies ] />
```

Inserts a table with the valuation of the controls that apply from the evaluation profile "code", in the requested domains, down to the requested depth.

### 13.1 Attributes

attribute	contents	default
<b>code</b>	code of the security profile	n.a.
<b>control</b>	list of control codes	all
<b>depth</b>	depth to list	all
<b>domain</b>	list of security domains	all
<b>phase</b>	list of phases	all
<b>mode</b>	if “maturity” Pilar prints the maturity level; otherwise, the degree of fulfilment	degree of fulfilment
<b>questions</b>	boolean: yes / no, true / false	no
<b>safeguards</b>	boolean: yes / no, true / false	no
<b>applies</b>	only prints the controls that apply, or not: yes – only those that apply no – only those that do not apply all – all	yes