PILAR: Personalization

13.8.2016

Contenido

1 Introduction

PILAR is distributed with some configuration files, copied onto the directory where you install it. Configuration files have an extension .car.

From this file there are pointers to other configuration files.

2 Configuration file (.car)

The file is plain text, and you may edit to change language and/or directories, if you change the standard installation.

You may tune it:

- add an icon of yourself
- add a splash screen
- change character separator for CSV files
- tune default asset layers, and administrative data
- adapt marking labels
- add new asset classes, new threats
- add new or replace valuation criteria
- adapt threat profile
- …

2.1 Directories

PILAR defines a HOME directory that, by default, is the directory that contains the configuration file (.car). You may define another directory

home= …

Other directories

library= … path relative to home …

Configuration details.

help= … path relative to home …

Help files.

info= … path relative to home

Information about asset classes, threats, …

util= … path relative to home …

Interface texts in different languages.

imgs= … path relative to util

Interface icons.

tpl = … path relative to home

Templates for reports.

2.2 Definitions

You may define labels; that is, assign a text to a label, and later reuse it.

PILAR has a built in definition for HOME, that stands for the directory containing the .car file.

Example. To allocate a number of personalization files in a project directory

```
PROJECT= $home$/project_ext
```

profile= $PROJECT$/my_profile.evl

reports= $PROJECT/my_reports.xml

2.3 Other elements

tool= pilar
version= 6.2
Informative.

locale= en_GB
Localization
See http://www.oracle.com/us/technologies/java/locale-140624.html

csv.separator= ;
Used when generating CSV files.

licensee_icon=
You may specify your own icon.

splash=
You may specify a splash screen, to be shown while launching.

model_info= info_en
Points to a file in the library directory.
See "Info" section below.

marking= marking_en.xml
Points to a file in the library directory where you specify that confidentiality marking for your organization. See "marking" section below.

levels-values= levels.xml
Points to a file in the library directory where it is established the mapping between qualitative levels and quantitative values.

extensions= ext_classes.lle
There may be several of this, each point to a file to be loaded from the library directory. See "extensions" below.

criteria= criteria_en.xml

Points to a file in the library directory where rating criteria for assets are defined, replacing the standard ones.

warnings= warnings_en.xml

Points to a file in the library where there are the language-specific translations of a few waring messages.

tsv= 2016-06-28.xlsx

Points to a file in the library directory where the standard threat values are defined. See "TSV" below.

ignore= ignored.xml

Points to a file in library directory where you may instruct PILAR to ignore dimensions and threats.

Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<ignored-dimensions>
  <ignore D="es.V" />
</ignored-dimensions>

<ignored-threats>
  <ignore Z="N.*" />
  <ignore Z="N.1" />
  <ignore Z="N.2" />
</ignored-threats>
```

attacker= [EXT_L] External attackers (cyber)

tsv:EXT_L= tsv_log.xml, 2016-06-28.xlsx

Definitions for zones. See information on Zones.

profile= 27002_2013_2016-06-16_en.evl

Points to a file in the library directory. The file provides a security profile.

protections= COM_radio_406_en.kb

Points to a file in the library directory. The file provides protections.

reports= reports.xml

Points to a file in the library directory that lists predefined templates: the file, and the name for users. It may also include dependency on a security profile. The profile must be loaded for the report to be presented.

Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<reports>
 <template>
  <file>Risk_en_tpl.rtf</file>
  <name>Risk analysis</name>
 </template>
 <template>
  <file>Compliance_27000_2013_en_tpl.rtf</file>
```

```
  <name>Compliance ISO/IEC 27000 (2013)</name>
  <evl>27002:2013</evl>
 </template>
 <template>
  <file>Compliance_27000_2005_en_tpl.rtf</file>
  <name>Compliance ISO/IEC 27000 (2005)</name>
  <evl>27002:2005</evl>
 </template>
</reports>
```

# help, relative to "home"
help= help_en
help2= http://www.pilar-tools.com/doc/v62/help_en
HELP points to a file under home, where help files are located.
HELP2 provides an alternative URL on the web, to refer to if local help is not available.

texts= common_en.txt
Points to a file in the util directory that provides texts for the user interface.

phase.PILAR= PILAR
The name of the special phase PILAR.

3 Marking

`marking= marking_en.xml`

```
<marking>
  <mark C="TS">TOP SECRET</mark>
  <mark C="S">SECRET</mark>
  <mark C="C">CONFIDENTIAL</mark>
  <mark C="R">RESTRICTED</mark>
  <mark C="UC">UNCLASSIFIED</mark>
</marking>
```

4 User criteria

`criteria= criteria_en.xml`

Replaces default valuation criteria in PILAR.

```
<criteria lang="en">
  <default value="10">
    [10] Level 10
  </default>
  <default value="9">
    [9] Level 9
```

```
  </default>

  <criterion>
    [lro] Legal and Regulatory Obligations:
    <criterion value="9">
      [9.lro] is likely to lead to an exceptionally serious
breach of a legal or regulatory obligation
      </criterion>
    <criterion value="7">
      [7.lro] is likely to lead to a major breach of a legal
or regulatory obligation
      </criterion>
    <criterion value="5">
      [5.lro] is likely to lead to a breach of a legal or
regulatory obligation
      </criterion>
    <criterion value="3">
      [3.lro] is likely to lead to a minor / technical breach
of a legal or regulatory obligation
      </criterion>
    <criterion value="1">
      [1.lro] could cause a minor / technical breach of a
legal or regulatory obligation
      </criterion>
    </criterion>
```

5 Warnings

warnings= warnings_es.xml

```
<warnings>

  <warning C="1">
    Some assets are not valued
  </warning>

  <warning C="2">
    No ENS class domains.
  </warning>

  <warning C="3">
    No IP class domains.
  </warning>

  <warning C="4">
    No D.log class asset.
  </warning>

</warnings>
```

6 Model information

model_info= info_en.info

Provides

- standard items for administrative data, for project and families
- standard layers to organize assets

```
<info>
  <model>
    <key c="desc">description</key>
    <key c="resp">responsible</key>
    <key c="org">organisation</key>
    <key c="ver">version</key>
    <key c="date">date</key>
  </model>
  <asset>
    <key c="desc">description</key>
    <key c="resp">responsible</key>
  </asset>
  <asset family="essential">
    <key c="owner">owner</key>
  </asset>
  <asset family="HW, COM, SI, AUX">
    <key c="location">location</key>
  </asset>
  <asset family="HW, COM, SI, AUX, L, P">
    <key c="number">number</key>
  </asset>
</info>

<assets>
  <layer c="B">Essential assets</layer>
  <layer c="IS">Internal services</layer>
  <layer c="E">Equipment
    <group c="SW">Applications</group>
    <group c="HW">Hardware</group>
    <group c="COM">Communications</group>
    <group c="AUX">Other elements</group>
  </layer>
  <layer c="SS">Subcontracted services</layer>
  <layer c="L">Facilities</layer>
  <layer c="P">Personnel</layer>
</assets>
```

7 Library extensions

extensions= ext_classes.lle

Extends the elements of the library. Mostly trees, into which new branches and leaves are placed.

Extension files for new criteria are written in XML notation using the following syntax

```
file ::=
  <library-extension>
    { classes }0+
    { criteria }0+
    { threats }0+
  </library-extension>


// new classes of assets

classes ::=
  <classes [ under="..." ] >
    { class }0+
    { aggregate }0+
  </classes>

class ::=
  <class code="...">
    NAME
    { threat_spec }0+
  </class>

aggregate ::=
  <aggregate code="...">
    NAME
    { class_ref }0+
    { threat_spec }0+
  </aggregate>

class_ref ::=
  <class code="..." />

threat_spec ::=
  <threat Z="..." f="..." s="...">
    { deg_spec }0+
  </threat>

deg_spec ::=
  <set D="dimension" deg="..."/>


// new valuation criteria

criteria ::=
```

```
  <criteria under="..." >
    { reason }0+
  </criteria>

reason ::=
  <reason code="..."> NAME </reason>


// new threats

threats ::=
  <threats under="..." >
    { threat }0+
  </threats>

threat ::=
  <threat code="..."> NAME </threat>
```

7.1 New classes of assets

Extension files for new classes of assets are written in XML notation using the following syntax

```
 6  classes ::=
 7    <classes [ under="..." ] >
 8      { class }0+
 9    </classes>
10
11  class ::=
12    <class code="..."> NAME </class>
```

Each new class has a code and a name. The name is given in line 12. There are two options for the code

- if there is an attribute “under” in line 7, the attribute “under” and the attribute “code” in line 12 are concatenated to build the full code of the new class
- if there is no attribute “under”, the code is the one given in line 12

In the first case, the class identified by “under” must be already defined.

In any case, the code of the class is the full path, and the new code is located under the class that has a prefix code. Prefixes are created as needed.

It is easier to understand with an example that creates several classes:

```
<library-extension>

  <classes>
    <class code="a.b">class b under a</class>
    <class code="a.c">class c under a</class>
  </classes>

  <classes under="L">
```

```
    <class code="b">class b under L (locations)</class>
    <class code="c">class c under L (locations)</class>
  </classes>

</library-extension>
```

7.1.1 Aggregates

A new class may be an aggregation of other classes

```
  <classes under="HW">
    <aggregate code="pcd_ip">
      portable computer device interconnected
      <class code="HW.mobile" />
      <class code="arch.ip" />
      <class code="SW.std.av" />
      <class code="SW.std.os" />
      <class code="SW.std.bp.pkt" />
    </aggregate>
  </classes>
```

7.1.2 Threats

A new class or class aggregate may specify threats for the new class.

```
  <classes under="SW.std.os">
    <class code="android">
      android
      <threat Z="A.8" f="10.0" s="7d">
        <set D="D" deg="1.0" />
        <set D="I" deg="0.1" />
        <set D="C" deg="0.5" />
      </threat>
    </class>
  </classes>
```

7.2 New criteria for valuation

Extension files for new criteria are written in XML notation using the following syntax

```
 6  criteria ::=
 7    <criteria under="..." >
 8      { reason }0+
 9    </criteria>
10
11  reason ::=
12    <reason code="..."> NAME </reason>
```

Each new criterion has a code (given in line 12), and a name (given in line 12). The new criterion is allocated under the criterion given in line 7, that must exist.

The code in line 12 may start with a dot (“.”), then the full code is the concatenation of the attribute “under” and the attribute “code”.

The following example

- defines 5 personalized levels, allocated under standard PILAR major headings
- defines 5 criteria for confidentiality, under each of the personalized levels
- for instance, there is a criterion coded [my-2.C]; that is my confidentiality criteria for level 2

```
<library-extension>

  <criteria under="7">
    <reason code="my-5">My Organisation - Level 5</reason>
  </criteria>
  <criteria under="5">
    <reason code="my-4">My Organisation - Level 4</reason>
  </criteria>
  <criteria under="3">
    <reason code="my-3">My Organisation - Level 3</reason>
  </criteria>
  <criteria under="2">
    <reason code="my-2">My Organisation - Level 2</reason>
  </criteria>
  <criteria under="0">
    <reason code="my-1">My Organisation - Level 1</reason>
  </criteria>

  <criteria under="my-5">
    <reason code=".C">
     CONFIDENTIAL:
     Information regarded as highly important and critical.
     Only a few number of people is allowed to know it.
    </reason>
  </criteria>

  <criteria under="my-4">
    <reason code=".C">
     RESTRICTED:
     Its revelation to unauthorised people
     would cause severe economic problems, or bad publicity.
    </reason>
  </criteria>

  <criteria under="my-3">
    <reason code=".C">
     RESTRICTED:
     Its revelation to unauthorised people
     would cause economic problems, bad publicity,
     or lead to a breach of a legal obligation..
    </reason>
```

```
  </criteria>

  <criteria under="my-2">
    <reason code=".C">
      INTERNAL:
      Information for internal use only:
      it is not expected to be outside the organisation.
    </reason>
  </criteria>

  <criteria under="my-1">
    <reason code=".C">
      PUBLIC:
      Information any one can access to (e.g. web site).
    </reason>
  </criteria>

</library-extension>
```

7.3 New threats

Extension files for new threats are written in XML notation using the following syntax

```
 6  threats ::=
 7    <threats under="..." >
 8      { threat }0+
 9    </threats>
10
11  threat ::=
12    <threat code="..."> NAME </threat>
```

Each new threat has a code (given in line 12), and a name (given in line 12). The new threat is allocated under the threat given in line 7, that must exist.

The code in line 12 may start with a dot (“.”), then the full code is the concatenation of the attribute “under” and the attribute “code”.

The following example

- defines several threats under the standard [N.*];
  the codes are [N.*.1], ...
- defines several threats under the standard [A.26];
  the codes are [beat], ...

```
<library-extension>

  <threats under="N.*">
    <threat code=".1">Storms</threat>
    <threat code=".2">Thunderstorms and Lightning</threat>
    <threat code=".3">Hurricanes</threat>
    <threat code=".4">Earthquakes</threat>
    <threat code=".5">Tornadoes</threat>
    <threat code=".6">Cyclones</threat>
    <threat code=".7">Landslide and mudslide</threat>
```

```
    <threat code=".8">Meteorites</threat>
    <threat code=".9">Tsunamis</threat>
    <threat code=".10">Winter storms and extreme cold</threat>
    <threat code=".11">Extreme heat</threat>
    <threat code=".12">Volcanoes</threat>
  </threats>

  <threats under="A.26">
    <threat code="brute">Beating / blowing</threat>
    <threat code="bomb">Bomb</threat>
    <threat code="terror">Terrorism</threat>
  </threats>

</library-extension>
```

8 Threat standard values (.tsv)

`tsv= 2013-07-27.tsv`

Provides

- standard threats for asset classes
- standard degradation and likelihood values for threats on assets

PILAR accepts two formats: XML and excel.

8.1 XML format

```
<threat-standard-values>

  <family F="HW.host">
    <threat f="0.1" s="15d" Z="E.25">
      <set D="D" deg="1.0"/>
      <set D="C" deg="1.0"/>
    </threat>
    <threat f="1.0" s="1h" Z="A.7">
      <set D="D" deg="0.01"/>
      <set D="I" deg="0.01"/>
      <set D="C" deg="0.1"/>
    </threat>
    <threat f="0.1" s="15d" Z="A.25">
      <set D="D" deg="1.0"/>
      <set D="C" deg="1.0"/>
    </threat>
  </family>
```

You may specify one or more asset classes at attribute “F” for tag “family”. Within that tag you may refer to threats that will apply to that family.

For tag “threat” you may specify

- one or more threats in attribute “Z”
- a likelihood, by means of an annual frequency
- a number of degradations on different dimensions, with sub-tag “set”. Degradation is a percent expressed a a real number 0.0 – 10.0
- an interruption step, “s”

Remember that asset classes are in a hierarchical tree; therefore, threats on a class apply to all its descendants, unless some descendant refines the values, hiding the ancestor.

The names of the dimensions of security are as Spanish acronyms, though you may use language specific names

dimension	es (default)	es (explicit)	en (explicit)	it (explicit)
availability	D	es.D	en.A	it.D
integrity	I	es.I	en.I	it.I
confidentiality	C	es.C	en.C	it.C
authenticity	A	es.A	en.Auth	it.A
accountability	T	es.T	en.Acc	it.T

8.2 Excel format

It can be prepared using an excel editor that exports in .xlsx format.

For example:

	A	B	C	D	E	F	G	H	I	J	K
1	app	family	threat	likely	step	D=en:A	D=en:I	D=en:C	D=en:Auth	D=en:Acc	D=en:V
2		D	E.15	1			1%				
3		D	E.18	1	1d	1%					
4		D	E.19	1				10%			
5		D	A.5	10			10%	50%	100%		
6		D	A.6	10	1d	1%	10%	50%			
7		D	A.11	100			10%	50%			
8		D.conf	E.15	0							
9		D.conf	E.19	0							
10		D.conf	E.4	1			1%				
11		D.conf	A.6	0							
12		D.conf	A.11	0							
13		D.conf	A.4	10	1d	10%	10%	10%			
14		D.log	E.19	0							

1st row is important in order to establish the content of each column. Each row is self-contained.

app

This column is used to comment out rows that we want to retain for whatever reason, but we do not want to apply.

Empty cells make the row applicable. Type “no” to skip the row.

family

Specifies the class of asset to which the row applies.

threat

Specifies the threat to which the row applies.

likely

Specifies potential likelihood: likelihood before safeguards are applied. The cell specifies the expected annual rate of occurrence (ARO).

step

Specifies the interruption caused by the threat.

D=dimension

Specifies the degradation caused by the threat on the referred dimension. The dimension is specified by language:acronym.

The degradation is specified as a percentage.

9 Ignored threats

`ignore= ignored_threats.xml`

Forces PILAR to ignore some threats.

```
<ignored-threats>
  <ignore Z="I.3" />
  <ignore Z="I.4" />
  <ignore Z="I.11" />
</ignored-threats>
```

10 Security profiles (.evl)

`profile= ens_2013-09-25_en.evl`

11 Protections (.kb)

`protections= COM_radio_406_en.kb`

12 XML notation

The XML syntax is presented using a variant of BNF notation, namely:

notation	meaning
`{ x }0+`	stands for zero or more occurrences of “x”
`{ x }1+`	stands for one or more occurrences of “x”

`[ x ]`	stands for zero or one occurrence of “x”; that is, “x” is optional