

RIA

Généré par Doxygen 1.8.15

1 Page principale	1
1.1 Description	1
1.2 schéma	1
2 RIA	3
3 Liste des choses à faire	5
4 Index des espaces de nommage	7
4.1 Paquetages	7
5 Index des fichiers	9
5.1 Liste des fichiers	9
6 Documentation des espaces de nommage	11
6.1 Référence de l'espace de nommage RIA	11
6.1.1 Documentation des fonctions	11
6.1.1.1 affiche_etat()	11
6.1.1.2 Close_env()	12
6.1.1.3 credit()	12
6.1.1.4 Init_env()	12
6.1.1.5 mon_IA()	12
6.1.2 Documentation des variables	13
6.1.2.1 action	13
6.1.2.2 args	13
6.1.2.3 default	13
6.1.2.4 dest	13
6.1.2.5 False	13
6.1.2.6 help	13
6.1.2.7 parser	13
6.2 Référence de l'espace de nommage test_json	14
6.2.1 Documentation des fonctions	14
6.2.1.1 Export_certfr_json()	14
6.2.2 Documentation des variables	14
6.2.2.1 html	14
6.2.2.2 MaBdd	14
7 Documentation des fichiers	15
7.1 Référence du fichier README.md	15
7.2 Référence du fichier RIA.py	15
7.2.1 Description détaillée	16
7.3 Référence du fichier test_json.py	16
Index	17

Chapitre 1

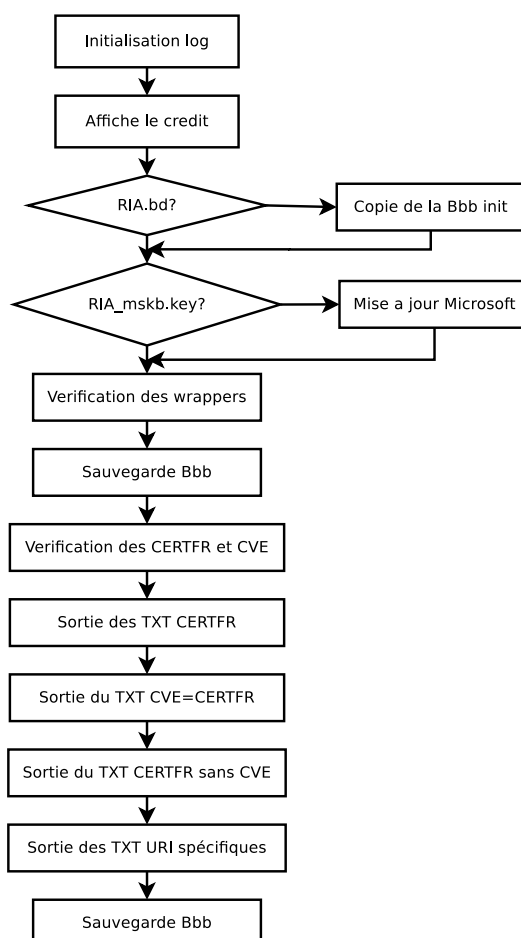
Page principale

1.1 Description

Télécharge et complète automatiquement les bulletins CERTFR avec les CVE/cpe
Si possible :

- les KB microsoft
- les informations éditeurs

1.2 schéma



Chapitre 2

RIA

Recherche Automatisé d'information

But

Apprendre le python.

Jouer avec du Json,XML,API WEB,RE, Sqlite, créer une class...

Compléter un bulletin du CERTFR [1](#) avec le plus d'informations possibles

Le code est donc pas propre ou bien optimisé.

Test sous :

- FEDORA 31 x64, Python 3.7.6
- WINDOWS 1903 x64, Python 3.8.1

Actuellement

- Téléchargement automatique des CERTFR et CVE
- Lecture des bulletins CERTFR depuis archive rar [1](#)
- Lecture des cve depuis les json [2](#)
- Gestion en SQLITE3
- Liste des KB via **API** microsoft [4](#)
- Wrapper les sites d'éditeurs pour trouver les CVE
- Ajouter de l'aide grace aux **Mogs** (ajout manuel)
- Utilisation des Docstring PEP 257 [3](#)
- Sortie en Json

Todo

- nettoyer le code
- Ajouter au Wrapper
- Ajout un commandline "--force" , "--help" ...
- ?..

Documentation

- Doxygen
- Graphviz
- Textlive pour LATEX to PDF

Dépendance PIP

- tqdm
- requests
- BeautifulSoup4

Références

Chapitre 3

Liste des choses à faire

Membre [RIA.mon_IA](#) ()

simplifier les répétitions

Fichier [RIA.py](#)

utiliser les best practice Python

ajouter des options en cmd `–forceupdate –help ...`

Chapitre 4

Index des espaces de nommage

4.1 Paquetages

Liste des paquetages avec une brève description (si disponible) :

RIA	11
test_json	14

Chapitre 5

Index des fichiers

5.1 Liste des fichiers

Liste de tous les fichiers avec une brève description :

RIA.py		
	Le coeur	15
test_json.py	16

Chapitre 6

Documentation des espaces de nommage

6.1 Référence de l'espace de nommage RIA

Fonctions

- `def credit ()`
Affiche simplement le crédit.
- `def Init_env ()`
initialisation des variables
- `def mon_IA ()`
le coeur du script
- `def Close_env ()`
- `def affiche_etat ()`

Variables

- `parser = argparse.ArgumentParser(description='Les options')`
Brief la ligne de commande.
- `dest`
- `action`
- `default`
- `False`
- `help`
- `args = parser.parse_args()`

6.1.1 Documentation des fonctions

6.1.1.1 `affiche_etat()`

```
def RIA.affiche_etat ( )
```

6.1.1.2 Close_env()

```
def RIA.Close_env ( )
```

6.1.1.3 credit()

```
def RIA.credit ( )
```

Affiche simplement le crédit.

Voici le graphe des appelants de cette fonction :



6.1.1.4 Init_env()

```
def RIA.Init_env ( )
```

initialisation des variables

6.1.1.5 mon_IA()

```
def RIA.mon_IA ( )
```

le coeur du script

A faire simplifier les répétitions

Voici le graphe d'appel pour cette fonction :



6.1.2 Documentation des variables

6.1.2.1 action

```
RIA.action
```

6.1.2.2 args

```
RIA.args = parser.parse_args()
```

6.1.2.3 default

```
RIA.default
```

6.1.2.4 dest

```
RIA.dest
```

6.1.2.5 False

```
RIA.False
```

6.1.2.6 help

```
RIA.help
```

6.1.2.7 parser

```
RIA.parser = argparse.ArgumentParser(description='Les options')
```

Brief la ligne de commande.

6.2 Référence de l'espace de nommage test_json

Fonctions

— def `Export_certfr_json` (*Outname*, *sql*)

Variables

— `MaBdd` = `C_sql()`
— list `html` = []

6.2.1 Documentation des fonctions

6.2.1.1 `Export_certfr_json()`

```
def test_json.Export_certfr_json (  
    Outname,  
    sql )
```

6.2.2 Documentation des variables

6.2.2.1 `html`

```
list test_json.html = []
```

6.2.2.2 `MaBdd`

```
test_json.MaBdd = C_sql()
```

Chapitre 7

Documentation des fichiers

7.1 Référence du fichier README.md

7.2 Référence du fichier RIA.py

Le coeur.

Espaces de nommage

— [RIA](#)

Fonctions

```
— def RIA.credit ()  
    Affiche simplement le crédit.  
— def RIA.Init\_env ()  
    initialisation des variables  
— def RIA.mon\_IA ()  
    le coeur du script  
— def RIA.Close\_env ()  
— def RIA.affiche\_etat ()
```

Variables

```
— RIA.parser = argparse.ArgumentParser(description='Les options')  
    Brief la ligne de commande.  
— RIA.dest  
— RIA.action  
— RIA.default  
— RIA.False  
— RIA.help  
— RIA.args = parser.parse_args()
```

7.2.1 Description détaillée

Le coeur.

Auteur

Frack113

Date

14/04/2020 Recherche d'Informations Automatisées

A faire utiliser les best practice Python
ajouter des options en cmd `-forceupdate -help ...`

7.3 Référence du fichier test_json.py

Espaces de nommage

— `test_json`

Fonctions

— `def test_json.Export_certfr_json (Outname, sql)`

Variables

— `test_json.MaBdd = C_sql()`
— `list test_json.html = []`

Index

- action
 - RIA, [13](#)
- affiche_etat
 - RIA, [11](#)
- args
 - RIA, [13](#)
- Close_env
 - RIA, [11](#)
- credit
 - RIA, [12](#)
- default
 - RIA, [13](#)
- dest
 - RIA, [13](#)
- Export_certfr_json
 - test_json, [14](#)
- False
 - RIA, [13](#)
- help
 - RIA, [13](#)
- html
 - test_json, [14](#)
- Init_env
 - RIA, [12](#)
- MaBdd
 - test_json, [14](#)
- mon_IA
 - RIA, [12](#)
- parser
 - RIA, [13](#)
- README.md, [15](#)
- RIA, [11](#)
 - action, [13](#)
 - affiche_etat, [11](#)
 - args, [13](#)
 - Close_env, [11](#)
 - credit, [12](#)
 - default, [13](#)
 - dest, [13](#)
 - False, [13](#)
 - help, [13](#)
 - Init_env, [12](#)
 - mon_IA, [12](#)
 - parser, [13](#)
 - RIA.py, [15](#)
 - test_json, [14](#)
 - Export_certfr_json, [14](#)
 - html, [14](#)
 - MaBdd, [14](#)
 - test_json.py, [16](#)