

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 classes	9
5.1 Liste des classes	9
6 Index des fichiers	11
6.1 Liste des fichiers	11
7 Documentation des espaces de nommage	13
7.1 Référence de l'espace de nommage RIA	13
7.1.1 Documentation des fonctions	13
7.1.1.1 credit()	13
7.1.1.2 mon_script()	14
7.2 Référence de l'espace de nommage RIA_class	14
7.3 Référence de l'espace de nommage RIA_mskb	14
7.4 Référence de l'espace de nommage RIA_out	14
7.5 Référence de l'espace de nommage RIA_sql	15
7.6 Référence de l'espace de nommage RIA_wrapper	15
7.7 Référence de l'espace de nommage test_json	15
7.7.1 Documentation des fonctions	15
7.7.1.1 Export_certfr_json()	15
7.7.2 Documentation des variables	15
7.7.2.1 html	15
7.7.2.2 MaBdd	15
8 Documentation des classes	17
8.1 Référence de la classe RIA_class.C_certfr	17
8.1.1 Description détaillée	18
8.1.2 Documentation des constructeurs et destructeur	18
8.1.2.1 __init__()	18
8.1.3 Documentation des fonctions membres	19
8.1.3.1 decode_file()	19
8.1.3.2 decode_link()	19
8.1.3.3 encode_file()	19
8.1.3.4 encode_link()	20
8.1.3.5 reset()	20

8.1.3.6 set_crc()	20
8.1.4 Documentation des données membres	20
8.1.4.1 crc	20
8.1.4.2 dateOrigine	21
8.1.4.3 dateUpdate	21
8.1.4.4 file	21
8.1.4.5 link	21
8.1.4.6 New	21
8.1.4.7 nom	21
8.1.4.8 obj	22
8.2 Référence de la classe RIA_class.C_cpe	22
8.2.1 Description détaillée	23
8.2.2 Documentation des constructeurs et destructeur	23
8.2.2.1 __init__()	23
8.2.3 Documentation des fonctions membres	23
8.2.3.1 reset()	24
8.2.3.2 set_crc()	24
8.2.4 Documentation des données membres	24
8.2.4.1 conf	24
8.2.4.2 cpe23Uri	24
8.2.4.3 crc	24
8.2.4.4 cve	25
8.2.4.5 id	25
8.2.4.6 New	25
8.2.4.7 operateur	25
8.2.4.8 versionEndExcluding	25
8.2.4.9 versionEndIncluding	25
8.2.4.10 versionStartExcluding	26
8.2.4.11 versionStartIncluding	26
8.2.4.12 vulnerable	26
8.3 Référence de la classe RIA_class.C_cve	26
8.3.1 Description détaillée	27
8.3.2 Documentation des constructeurs et destructeur	27
8.3.2.1 __init__()	27
8.3.3 Documentation des fonctions membres	28
8.3.3.1 reset()	28
8.3.3.2 set_crc()	28
8.3.4 Documentation des données membres	28
8.3.4.1 crc	28
8.3.4.2 cvssV2	28
8.3.4.3 cvssV2base	29
8.3.4.4 cvssV3	29

8.3.4.5 cvssV3base	29
8.3.4.6 dateOrigine	29
8.3.4.7 dateUpdate	29
8.3.4.8 id	29
8.3.4.9 New	30
8.4 Référence de la classe RIA_mskb.C_mskb	30
8.4.1 Description détaillée	31
8.4.2 Documentation des constructeurs et destructeur	31
8.4.2.1 __init__()	31
8.4.3 Documentation des fonctions membres	31
8.4.3.1 Check_Mskb_Update()	32
8.4.3.2 get_info_certfr()	32
8.4.3.3 reset_db()	33
8.4.3.4 update_all_info()	33
8.4.3.5 update_all_url()	34
8.4.3.6 write_cve_kb()	34
8.4.3.7 write_product()	35
8.4.4 Documentation des données membres	36
8.4.4.1 api_key	36
8.4.4.2 api_type	36
8.4.4.3 api_url	36
8.4.4.4 header	36
8.4.4.5 MaBdd	36
8.5 Référence de la classe RIA_out.C_out	37
8.5.1 Description détaillée	37
8.5.2 Documentation des constructeurs et destructeur	38
8.5.2.1 __init__()	38
8.5.3 Documentation des fonctions membres	38
8.5.3.1 CERT_to_STR()	38
8.5.3.2 Export_certfr_json()	39
8.5.3.3 MS_to_STR()	39
8.5.3.4 tab2_to_txt()	40
8.5.3.5 URI_to_FILE()	40
8.5.3.6 Write_CERTFR()	41
8.5.4 Documentation des données membres	41
8.5.4.1 Ksoft	42
8.5.4.2 MaBdd	42
8.6 Référence de la classe RIA_sql.C_sql	42
8.6.1 Description détaillée	43
8.6.2 Documentation des constructeurs et destructeur	44
8.6.2.1 __init__()	44
8.6.3 Documentation des fonctions membres	44

8.6.3.1 clean_new()	44
8.6.3.2 clean_tmp()	44
8.6.3.3 close_db()	45
8.6.3.4 flush_tmp()	45
8.6.3.5 flush_url()	45
8.6.3.6 get_all_certfr_by_cve()	46
8.6.3.7 get_all_cpe_certfr()	46
8.6.3.8 get_all_cpe_uri()	46
8.6.3.9 get_all_cve_certfr()	47
8.6.3.10 get_all_cve_orphan()	47
8.6.3.11 get_all_new_certfr()	48
8.6.3.12 get_all_orphan()	48
8.6.3.13 get_certfr()	48
8.6.3.14 get_Info_date()	49
8.6.3.15 get_max_lg_uri_cpe()	49
8.6.3.16 get_orphan_by_obj()	50
8.6.3.17 get_sc()	50
8.6.3.18 load_mogs()	50
8.6.3.19 save_db()	51
8.6.3.20 set_Info_date()	51
8.6.3.21 write_certfr_cve()	52
8.6.3.22 write_certfr_tmp()	52
8.6.3.23 write_cpe_tmp()	53
8.6.3.24 write_cve_tmp()	53
8.6.3.25 write_sc()	53
8.6.4 Documentation des données membres	54
8.6.4.1 Localdb	54
8.6.4.2 moncur	54
8.7 Référence de la classe RIA_wrapper.C_wrapper	54
8.7.1 Description détaillée	55
8.7.2 Documentation des constructeurs et destructeur	55
8.7.2.1 __init__()	56
8.7.3 Documentation des fonctions membres	56
8.7.3.1 Check_ALL_Wapper_Update()	56
8.7.3.2 Check_Gitlab()	57
8.7.3.3 Check_Kaspersky()	57
8.7.3.4 Check_Redhat()	58
8.7.3.5 check_regex()	59
8.7.3.6 Check_Ubuntu()	60
8.7.3.7 Check_Xen()	61
8.7.3.8 Download_Certfr()	61
8.7.3.9 Download_CVE()	62

8.7.3.10 Flush_cve()	62
8.7.3.11 Load_TAR_certfr()	62
8.7.3.12 Load_ZIP_cve()	64
8.7.3.13 Read_wrapper_info()	65
8.7.3.14 Reset_New()	65
8.7.3.15 Search_re()	66
8.7.3.16 Url_down_file()	66
8.7.3.17 Url_is_updated()	67
8.7.3.18 write_url_cve()	68
8.7.3.19 Write_wrapper_info()	69
8.7.4 Documentation des données membres	69
8.7.4.1 MaBdd	69
8.7.4.2 mylog	69
8.8 Référence de la classe RIA_wrapper.C_wrapper_info	70
8.8.1 Description détaillée	70
8.8.2 Documentation des constructeurs et destructeur	71
8.8.2.1 __init__()	71
8.8.3 Documentation des données membres	71
8.8.3.1 Date	71
8.8.3.2 Fichier	71
8.8.3.3 Module	71
8.8.3.4 New	71
8.8.3.5 Regex	72
8.8.3.6 Rep	72
8.8.3.7 S_Url	72
8.8.3.8 Taille	72
8.8.3.9 Url	72
9 Documentation des fichiers	73
9.1 Référence du fichier package/RIA_class.py	73
9.1.1 Description détaillée	73
9.2 Référence du fichier package/RIA_mskb.py	73
9.2.1 Description détaillée	74
9.3 Référence du fichier package/RIA_out.py	75
9.3.1 Description détaillée	75
9.4 Référence du fichier package/RIA_sql.py	75
9.4.1 Description détaillée	75
9.5 Référence du fichier package/RIA_wrapper.py	76
9.5.1 Description détaillée	76
9.6 Référence du fichier README.md	76
9.7 Référence du fichier RIA.py	76
9.7.1 Description détaillée	77

9.8 Référence du fichier test_json.py	77
Index	79

Chapitre 1

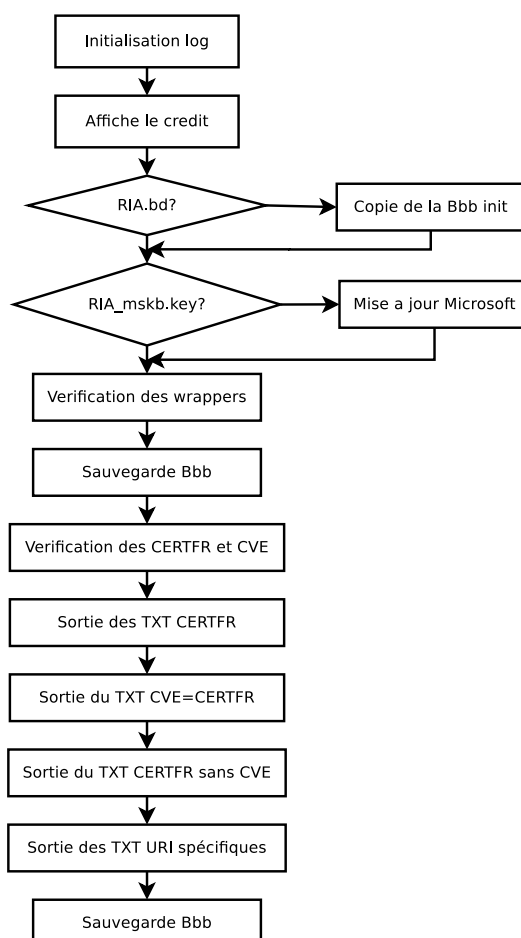
Page principale

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_script ()`

simplifier les répétitions

Fichier `RIA.py`

utiliser les best practice Python

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

Membre `RIA_mskb.C_mskb.get_info_certfr (self, certfr)`

retravailler la requête SQL

Fichier `RIA_mskb.py`

optimiser le code

peut être une "class" Wsus avec le cab <http://go.microsoft.com/fwlink/?linkid=74689>

Membre `RIA_out.C_out.MS_to_STR (self, Nom, tab)`

gérer la taille dynamique des colonnes

Membre `RIA_sql.C_sql.get_Info_date (self, quoi)`

changer le nom des champs en plus explicite

Membre `RIA_sql.C_sql.get_max_lg_uri_cpe (self, certfr)`

gérer les erreurs

Membre `RIA_wrapper.C_wrapper.Load_TAR_certfr (self, file)`

nettoyer le code

Membre `RIA_wrapper.C_wrapper.Load_ZIP_cve (self, file)`

nettoyer le code

Chapitre 4

Index des espaces de nommage

4.1 Paquetages

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

RIA	13
RIA_class	14
RIA_mskb	14
RIA_out	14
RIA_sql	15
RIA_wrapper	15
test_json	15

Chapitre 5

Index des classes

5.1 Liste des classes

Liste des classes, structures, unions et interfaces avec une brève description :

RIA_class.C_certfr	
Manipulation des CERTFR	17
RIA_class.C_cpe	
Manipulation CPE	22
RIA_class.C_cve	
Manipulation des CVE	26
RIA_mskb.C_mskb	
Gestion API Microsoft	30
RIA_out.C_out	
Gestion des sorties	37
RIA_sql.C_sql	
Manipulation de la BDD	42
RIA_wrapper.C_wrapper	
Class pour le Wrapper	54
RIA_wrapper.C_wrapper_info	
Class Objet info Wrapper	70

Chapitre 6

Index des fichiers

6.1 Liste des fichiers

Liste de tous les fichiers avec une brève description :

RIA.py		
Le coeur	76
test_json.py	77
package/RIA_class.py		
Gestion des CERTFR,CVE et CPE	73
package/RIA_mskb.py		
La gestion de l'API Microsoft	73
package/RIA_out.py		
Gestion des sorties	75
package/RIA_sql.py		
La gestion de la base de données	75
package/RIA_wrapper.py		
La gestion des wrappers Internet	76

Chapitre 7

Documentation des espaces de nommage

7.1 Référence de l'espace de nommage RIA

Fonctions

- def `credit` ()
Affiche simplement le crédit.
- def `mon_script` ()
Core du script.

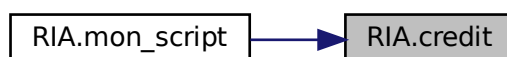
7.1.1 Documentation des fonctions

7.1.1.1 `credit()`

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

Affiche simplement le crédit.

Voici le graphe des appelants de cette fonction :



7.1.1.2 mon_script()

```
def RIA.mon_script ( )
```

Core du script.

le coeur du script

A faire simplifier les répétitions

Voici le graphe d'appel pour cette fonction :



7.2 Référence de l'espace de nommage RIA_class

Classes

- class [C_certfr](#)
Manipulation des CERTFR.
- class [C_cpe](#)
Manipulation CPE.
- class [C_cve](#)
Manipulation des CVE.

7.3 Référence de l'espace de nommage RIA_mskb

Classes

- class [C_mskb](#)
Gestion API Microsoft.

7.4 Référence de l'espace de nommage RIA_out

Classes

- class [C_out](#)
Gestion des sorties.

7.5 Référence de l'espace de nommage RIA_sql

Classes

- class `C_sql`
Manipulation de la BDD.

7.6 Référence de l'espace de nommage RIA_wrapper

Classes

- class `C_wrapper`
Class pour le Wrapper.
- class `C_wrapper_info`
Class Objet info Wrapper.

7.7 Référence de l'espace de nommage test_json

Fonctions

- def `Export_certfr_json` (Outname, sql)

Variables

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

7.7.1 Documentation des fonctions

7.7.1.1 Export_certfr_json()

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

7.7.2 Documentation des variables

7.7.2.1 html

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

7.7.2.2 MaBdd

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


Chapitre 8

Documentation des classes

8.1 Référence de la classe RIA_class.C_certfr

Manipulation des CERTFR.

Graphe de collaboration de RIA_class.C_certfr:

RIA_class.C_certfr
+ nom + obj + dateOrigine + dateUpdate + New + file + crc + link
+ __init__() + reset() + set_crc() + decode_file() + encode_file() + encode_link() + decode_link()

Fonctions membres publiques

- def `__init__` (self)
The constructor.
- def `reset` (self)
permet de remettre les variables à l'état initial
- def `set_crc` (self)

- *Calcule le CRC pour la clé UNIQUE SQL.*
- def `decode_file` (self)
décode le fichier base64
- def `encode_file` (self)
encode le fichier en base64
- def `encode_link` (self)
encode les liens en base64
- def `decode_link` (self)
décode les liens en base64

Attributs publics

- `nom`
le nom du bulletin
- `obj`
l'objet du bulletin
- `dateOrigine`
Date de création du bulletin.
- `dateUpdate`
Date de modification du bulletin.
- `New`
boolean 0 déjà traité , 1 nouveau
- `file`
Le bulletin complet.
- `crc`
Clé unique.
- `link`
Les liens dans le bulletin.

8.1.1 Description détaillée

Manipulation des CERTFR.

Python help

```
Class" qui représente un bulletin du CERTFR
```

8.1.2 Documentation des constructeurs et destructeur

8.1.2.1 `__init__()`

```
def RIA_class.C_certfr.__init__ (
    self )
```

The constructor.

Python help

```
le constructor
```

8.1.3 Documentation des fonctions membres

8.1.3.1 decode_file()

```
def RIA_class.C_certfr.decode_file (
    self )
```

décode le fichier base64

Python help

Décodage de file (base64 en BDD)

8.1.3.2 decode_link()

```
def RIA_class.C_certfr.decode_link (
    self )
```

décode les liens en base64

Avertissement

sûrement plus utile

Python help

Décodage des liens (base64 en BDD)

8.1.3.3 encode_file()

```
def RIA_class.C_certfr.encode_file (
    self )
```

encode le fichier en base64

Python help

Encodage en base64 file pour le stockage en BDD

8.1.3.4 encode_link()

```
def RIA_class.C_certfr.encode_link (
    self )
```

encode les liens en base64

Avertissement

sûrement plus utile

Python help

Encodage des liens (base64 en BDD)

8.1.3.5 reset()

```
def RIA_class.C_certfr.reset (
    self )
```

permet de remettre les variables à l'état initial

Python help

Remet à l'état d'origine les variables

8.1.3.6 set_crc()

```
def RIA_class.C_certfr.set_crc (
    self )
```

Calcule le CRC pour la clé UNIQUE SQL.

Python help

Calcule le Hkey UNIQUE en SHA1

8.1.4 Documentation des données membres

8.1.4.1 crc

RIA_class.C_certfr.crc

Clé unique.

8.1.4.2 `dateOrigine`

`RIA_class.C_certfr.dateOrigine`

Date de création du bulletin.

8.1.4.3 `dateUpdate`

`RIA_class.C_certfr.dateUpdate`

Date de modification du bulletin.

8.1.4.4 `file`

`RIA_class.C_certfr.file`

Le bulletin complet.

8.1.4.5 `link`

`RIA_class.C_certfr.link`

Les liens dans le bulletin.

8.1.4.6 `New`

`RIA_class.C_certfr.New`

boolean 0 déjà traité , 1 nouveau

8.1.4.7 `nom`

`RIA_class.C_certfr.nom`

le nom du bulletin

8.1.4.8 obj

`RIA_class.C_certfr.obj`

l'objet du bulletin

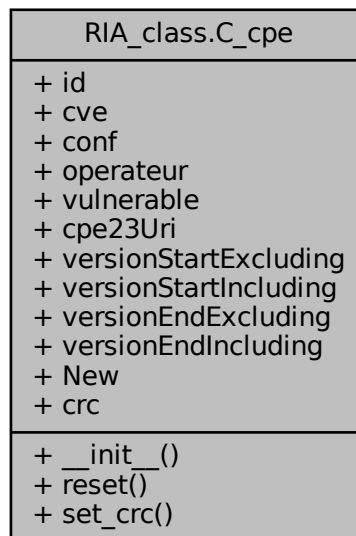
La documentation de cette classe a été générée à partir du fichier suivant :

— package/[RIA_class.py](#)

8.2 Référence de la classe `RIA_class.C_cpe`

Manipulation CPE.

Graphe de collaboration de `RIA_class.C_cpe`:



Fonctions membres publiques

- def `__init__` (self)
The constructor.
- def `reset` (self)
Remet les variables à l'état initial.
- def `set_crc` (self)
Calcule le CRC pour la clé UNIQUE SQL.

Attributs publics

- `id`
l'ID du CPE
- `cve`
Le cve de référence.
- `conf`
Nombre de configuration.
- `opérateur`
Opérateur OR ou AND.
- `vulnerable`
Si vulnérable ex Firefox sur windows firefox AND (false) Windows.
- `cpe23Uri`
l'URI en 2.3
- `versionStartExcluding`
Version de départ exclue.
- `versionStartIncluding`
Version de départ incluse.
- `versionEndExcluding`
Version de fin exclue.
- `versionEndIncluding`
Version de fin incluse.
- `New`
boolean 0 déjà traité , 1 nouveau
- `crc`
Clé unique.

8.2.1 Description détaillée

Manipulation CPE.

Python help

Class qui représente un CPE d'un CVE du NIST

8.2.2 Documentation des constructeurs et destructeur

8.2.2.1 `__init__()`

```
def RIA_class.C_cpe.__init__ (
    self )
```

The constructor.

Python help

le constructor

8.2.3 Documentation des fonctions membres

8.2.3.1 reset()

```
def RIA_class.C_cpe.reset (
    self )
```

Remet les variables à l'état initial.

Python help

Remet à l'état d'origine les variables

8.2.3.2 set_crc()

```
def RIA_class.C_cpe.set_crc (
    self )
```

Calcule le CRC pour la clé UNIQUE SQL.

Python help

Calcule Hkey UNIQUE en SHA1

8.2.4 Documentation des données membres

8.2.4.1 conf

```
RIA_class.C_cpe.conf
```

Nombre de configuration.

8.2.4.2 cpe23Uri

```
RIA_class.C_cpe.cpe23Uri
```

l'URI en 2.3

8.2.4.3 crc

```
RIA_class.C_cpe.crc
```

Clé unique.

8.2.4.4 `cve`

`RIA_class.C_cpe.cve`

Le cve de référence.

8.2.4.5 `id`

`RIA_class.C_cpe.id`

l'ID du CPE

8.2.4.6 `New`

`RIA_class.C_cpe.New`

boolean 0 déjà traité , 1 nouveau

8.2.4.7 `opérateur`

`RIA_class.C_cpe.opérateur`

Opérateur OR ou AND.

8.2.4.8 `versionEndExcluding`

`RIA_class.C_cpe.versionEndExcluding`

Version de fin exclue.

8.2.4.9 `versionEndIncluding`

`RIA_class.C_cpe.versionEndIncluding`

Version de fin incluse.

8.2.4.10 versionStartExcluding

`RIA_class.C_cpe.versionStartExcluding`

Version de départ exclue.

8.2.4.11 versionStartIncluding

`RIA_class.C_cpe.versionStartIncluding`

Version de départ incluse.

8.2.4.12 vulnerable

`RIA_class.C_cpe.vulnerable`

Si vulnérable ex Firefox sur windows firefox AND (false) Windows.

La documentation de cette classe a été générée à partir du fichier suivant :

— [package/RIA_class.py](#)

8.3 Référence de la classe `RIA_class.C_cve`

Manipulation des CVE.

Graphe de collaboration de `RIA_class.C_cve`:

RIA_class.C_cve
+ id + cvssV3 + cvssV3base + cvssV2 + cvssV2base + dateOrigine + dateUpdate + New + crc
+ __init__() + reset() + set_crc()

Fonctions membres publiques

- def `__init__` (self)
Le constructor.
- def `reset` (self)
Remet les variables à l'état initial.
- def `set_crc` (self)
Calcule le CRC pour la clé UNIQUE SQL.

Attributs publics

- `id`
L'id CVE.
- `cvssV3`
Le cvssV3.
- `cvssV3base`
La note de Base cvssV3.
- `cvssV2`
Le cvssV2.
- `cvssV2base`
La note de base cvssV2.
- `dateOrigine`
La date de création du CVE.
- `dateUpdate`
La date dernière modification du CVE.
- `New`
boolean 0 déjà traité , 1 nouveau
- `crc`
Clé unique.

8.3.1 Description détaillée

Manipulation des CVE.

Python help

```
Class qui représente un CVE du NIST
```

8.3.2 Documentation des constructeurs et destructeur

8.3.2.1 `__init__()`

```
def RIA_class.C_cve.__init__ (
    self )
```

Le constructor.

Python help

```
le constructor
```

8.3.3 Documentation des fonctions membres

8.3.3.1 reset()

```
def RIA_class.C_cve.reset (
    self )
```

Remet les variables à l'état initial.

Python help

Remet à l'état d'origine les variables

8.3.3.2 set_crc()

```
def RIA_class.C_cve.set_crc (
    self )
```

Calcule le CRC pour la clé UNIQUE SQL.

Python help

Calcule Hkey UNIQUE en SHA1

8.3.4 Documentation des données membres

8.3.4.1 crc

RIA_class.C_cve.crc

Clé unique.

8.3.4.2 cvssV2

RIA_class.C_cve.cvssV2

Le cvssV2.

8.3.4.3 `cvssV2base`

`RIA_class.C_cve.cvssV2base`

La note de base cvssV2.

8.3.4.4 `cvssV3`

`RIA_class.C_cve.cvssV3`

Le cvssV3.

8.3.4.5 `cvssV3base`

`RIA_class.C_cve.cvssV3base`

La note de Base cvssV3.

8.3.4.6 `dateOrigine`

`RIA_class.C_cve.dateOrigine`

La date de création du CVE.

8.3.4.7 `dateUpdate`

`RIA_class.C_cve.dateUpdate`

La date dernière modification du CVE.

8.3.4.8 `id`

`RIA_class.C_cve.id`

L'id CVE.

8.3.4.9 New

`RIA_class.C_cve.New`

boolean 0 déjà traité , 1 nouveau

La documentation de cette classe a été générée à partir du fichier suivant :

— package/[RIA_class.py](#)

8.4 Référence de la classe RIA_mskb.C_mskb

Gestion API Microsoft.

Graphe de collaboration de `RIA_mskb.C_mskb`:

RIA_mskb.C_mskb
+ api_key + api_type + api_url + header + MaBdd
+ __init__() + reset_db() + update_all_url() + write_product() + write_cve_kb() + update_all_info() + get_info_certfr() + Check_Mskb_Update()

Fonctions membres publiques

- def `__init__` (self, [MaBdd](#))
Constructors.
- def `reset_db` (self)
Efface totalement les tables.
- def `update_all_url` (self)
récupère toutes les URL
- def `write_product` (self, ProductID, Value)
sauvegarde en BDD un ProductID
- def `write_cve_kb` (self, ms_cve, fix_id, product, ms_url, fix_Supercedence, typekb)
sauvegarde en BDD un cve
- def `update_all_info` (self)
récupère toutes les informations
- def `get_info_certfr` (self, certfr)
Cherche les informations pour un CERTFR.
- def `Check_Mskb_Update` (self, date)
Vérifie la mise à jour.

Attributs publics

- `api_key`
la clé API
- `api_type`
le type de transaction
- `api_url`
L'url API.
- `header`
le header
- `MaBdd`
La Bdd.

8.4.1 Description détaillée

Gestion API Microsoft.

Python help

```
Class pour l'utilisation de l'api microsoft
https://portal.msrc.microsoft.com/fr-fr/developer
```

8.4.2 Documentation des constructeurs et destructeur**8.4.2.1 __init__()**

```
def RIA_mskb.C_mskb.__init__ (
    self,
    MaBdd )
```

Constructors.

Paramètres

<i>MaBdd</i>	un objet C_sql
--------------	----------------

Python help

```
Constructors
Mabdd est un objet C_sql déjà initialisé
il faut une clé API sauvegardée dans RIA_mskb.key
Les tables SQL manquantes sont créées automatiquement
```

8.4.3 Documentation des fonctions membres

8.4.3.1 Check_Mskb_Update()

```
def RIA_mskb.C_mskb.Check_Mskb_Update (
    self,
    date )
```

Vérifie la mise à jour.

Paramètres

<i>date</i>	la date "YYYYMMDD" à vérifier
-------------	-------------------------------

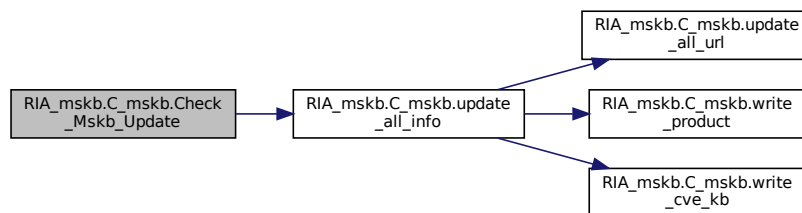
Renvoie

String d'informations

Python help

Vérifie s'il y des mises à jour par API

Voici le graphe d'appel pour cette fonction :



8.4.3.2 get_info_certfr()

```
def RIA_mskb.C_mskb.get_info_certfr (
    self,
    certfr )
```

Cherche les informations pour un CERTFR.

Paramètres

<i>certfr</i>	le nom du CERTFR
---------------	------------------

A faire retravailler la requête SQL

Python help

Renvoie une liste pour un nom de bulletin donné
certfr est une string avec le nom à chercher

8.4.3.3 reset_db()

```
def RIA_mskb.C_mskb.reset_db (
    self )
```

Efface totalement les tables.

Python help

Efface toutes les tables SQL

8.4.3.4 update_all_info()

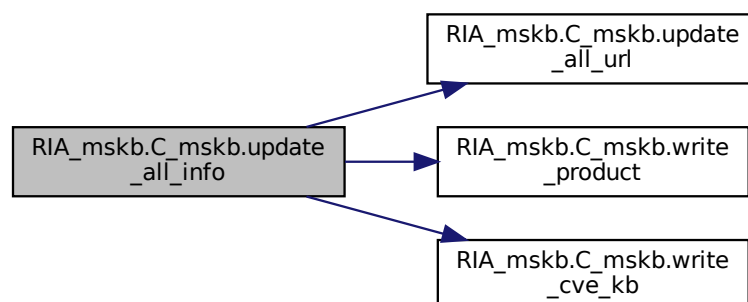
```
def RIA_mskb.C_mskb.update_all_info (
    self )
```

récupère toutes les informations

Python help

Vérifie toutes les pages d'informations

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.4.3.5 update_all_url()

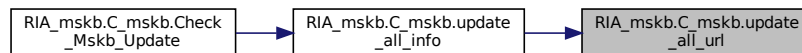
```
def RIA_mskb.C_mskb.update_all_url (
    self )
```

récupère toutes les URL

Python help

Récupère toutes les url à traiter

Voici le graphe des appelants de cette fonction :



8.4.3.6 write_cve_kb()

```
def RIA_mskb.C_mskb.write_cve_kb (
    self,
    ms_cve,
    fix_id,
    product,
    ms_url,
    fix_Supercedence,
    typekb )
```

sauvegarde en BDD un cve

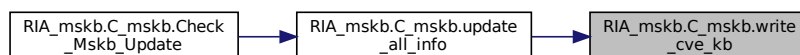
Paramètres

<i>ms_cve</i>	le CVE corrigé
<i>fix_id</i>	le numéro de KB
<i>product</i>	le ProductID
<i>ms_url</i>	l'URL pour plus d'informations
<i>fix_Supercedence</i>	le KB remplacé
<i>typekb</i>	le type de KB

Python help

Ecrit en BDD les informations d'une CVE MS

Voici le graphe des appelants de cette fonction :



8.4.3.7 write_product()

```
def RIA_mskb.C_mskb.write_product (
    self,
    ProductID,
    Value )
```

sauvegarde en BDD un ProductID

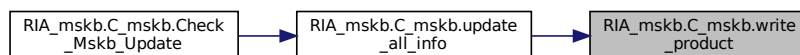
Paramètres

<i>ProductID</i>	le nmr de ref
<i>Value</i>	la valeur lisible

Python help

Inscrit en BDD le couple "ID:Nom en clair"

Voici le graphe des appelants de cette fonction :



8.4.4 Documentation des données membres

8.4.4.1 `api_key`

`RIA_mskb.C_mskb.api_key`

la clé API

8.4.4.2 `api_type`

`RIA_mskb.C_mskb.api_type`

le type de transaction

8.4.4.3 `api_url`

`RIA_mskb.C_mskb.api_url`

L'url API.

8.4.4.4 `header`

`RIA_mskb.C_mskb.header`

le header

8.4.4.5 `MaBdd`

`RIA_mskb.C_mskb.MaBdd`

La Bdd.

La documentation de cette classe a été générée à partir du fichier suivant :

— [package/RIA_mskb.py](#)

8.5 Référence de la classe RIA_out.C_out

Gestion des sorties.

Graphe de collaboration de RIA_out.C_out:

RIA_out.C_out
+ MaBdd + Ksoft
+ __init__() + MS_to_STR() + CERT_to_STR() + Write_CERTFR() + URI_to_FILE() + tab2_to_txt() + Export_certfr_json()

Fonctions membres publiques

- def `__init__` (self, `MaBdd`, `Ksoft`)
constructors
- def `MS_to_STR` (self, Nom, tab)
Une jolie sortie formatée des info Microsoft.
- def `CERT_to_STR` (self, Nom, tab)
Une jolie sortie formatée des info CERTFR.
- def `Write_CERTFR` (self, nom, rep)
Ecrit dans un fichier text les informations du bulletin.
- def `URI_to_FILE` (self, Nom, uri)
Ecrit un fichier avec bulletins et URI23 pour une recherche.
- def `tab2_to_txt` (self, Nom_sortie, tab)
Ecrit dans un fichier une liste à 2 champs.
- def `Export_certfr_json` (self, Outname, sql)
Export Json.

Attributs publics

- `MaBdd`
la Bdd via C_sql
- `Ksoft`

8.5.1 Description détaillée

Gestion des sorties.

Python help

8.5.2 Documentation des constructeurs et destructeur

8.5.2.1 `__init__()`

```
def RIA_out.C_out.__init__ (
    self,
    MaBdd,
    Ksoft )
```

constructors

Paramètres

<i>MaBdd</i>	C_sql
<i>Ksoft</i>	C_mskb

Python help

```
le constructor
MaBdd est un objet C_sql déjà existant
Ksoft est un objet C_mskb déjà existant
```

8.5.3 Documentation des fonctions membres

8.5.3.1 `CERT_to_STR()`

```
def RIA_out.C_out.CERT_to_STR (
    self,
    Nom,
    tab )
```

Une jolie sortie formatée des info CERTFR.

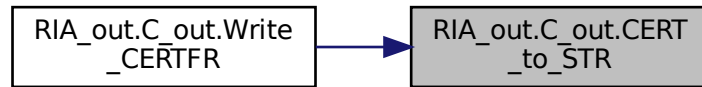
Paramètres

<i>Nom</i>	le nom du bulletin
<i>tab</i>	une liste

Python help

```
Ajoute à liste tab les informations CVE cpe
Nom string nom du Bulletin
```

Voici le graphe des appelants de cette fonction :



8.5.3.2 Export_certfr_json()

```
def RIA_out.C_out.Export_certfr_json (
    self,
    Outname,
    sql )
```

Export Json.

Paramètres

<i>Outname</i>	nom du fichier
<i>sql</i>	la fin de la requête

Python help

Export en json tous les bulletins requête SQL: WHERE Non {sql} dans le fichier 'Outname'
 Exemples :
 Un bulletin sql= '="CERTFR-2020-AVI-001"
 les 2020 sql= 'LIKE "%2020%'

8.5.3.3 MS_to_STR()

```
def RIA_out.C_out.MS_to_STR (
    self,
    Nom,
    tab )
```

Une jolie sortie formatée des info Microsoft.

Paramètres

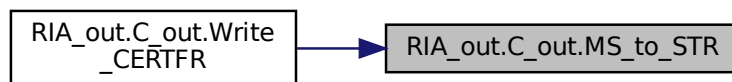
<i>Nom</i>	le nom du bulletin
<i>tab</i>	une liste

A faire gérer la taille dynamique des colonnes

Python help

ajoute à liste tab les informations Microsoft
Nom string avec le nom du bulletin

Voici le graphe des appelants de cette fonction :



8.5.3.4 tab2_to_txt()

```
def RIA_out.C_out.tab2_to_txt (
    self,
    Nom_sortie,
    tab )
```

Ecrit dans un fichier une liste à 2 champs.

Paramètres

<i>Nom_sortie</i>	nom du fichier
<i>tab</i>	la liste à écrire

Python help

Ecrit un fichier txt une liste 2 champs

8.5.3.5 URI_to_FILE()

```
def RIA_out.C_out.URI_to_FILE (
    self,
    Nom,
    uri )
```

Ecrit un fichier avec bulletins et URI23 pour une recherche.

Paramètres

<i>Nom</i>	recherche dans les objets et aussi le nom du fichier de sortie
<i>uri</i>	chaîne à chercher dans les uri23

Python help

Ecrit dans un fichier tous les bulletins avec 'Nom' dans l'objet
et toutes les uri23 SQL LIKE %uri%

8.5.3.6 Write_CERTFR()

```
def RIA_out.C_out.Write_CERTFR (
    self,
    nom,
    rep )
```

Ecrit dans un fichier text les informations du bulletin.

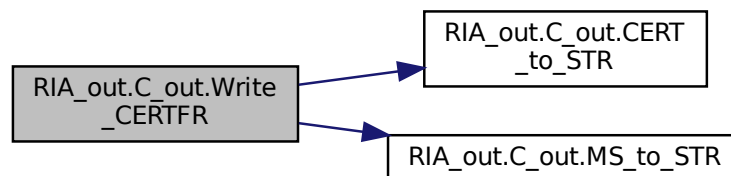
Paramètres

<i>nom</i>	le nom du bulletin
<i>rep</i>	répertoire de sortie

Python help

Ecrit les informations du bulletin 'nom' dans le répertoire txt/'rep'

Voici le graphe d'appel pour cette fonction :



8.5.4 Documentation des données membres

8.5.4.1 Ksoft

`RIA_out.C_out.Ksoft`

8.5.4.2 MaBdd

`RIA_out.C_out.MaBdd`

la Bdd via `C_sql`

La documentation de cette classe a été générée à partir du fichier suivant :

— [package/RIA_out.py](#)

8.6 Référence de la classe `RIA_sql.C_sql`

Manipulation de la BDD.

Graphe de collaboration de `RIA_sql.C_sql`:

RIA_sql.C_sql
+ Localdb + moncur
+ __init__() + close_db() + save_db() + get_Info_date() + set_Info_date() + clean_tmp() + clean_new() + write_sc() + get_sc() + write_certfr_tmp() et 16 de plus...

Fonctions membres publiques

- def `__init__` (self)
constructors
- def `close_db` (self)
fermeture
- def `save_db` (self)
Sauvegarde.
- def `get_Info_date` (self, quoi)
cherche une date dans la table information
- def `set_Info_date` (self, quoi, date)
écrit dans la table information.
- def `clean_tmp` (self)
efface les tables temporaires
- def `clean_new` (self)
Met New à "0".
- def `write_sc` (self, script)
Exécute un script sqlite sans retour.
- def `get_sc` (self, script)
Exécute un script sqlite.
- def `write_certfr_tmp` (self, monbul)
Ecrit en BDD un bulletin.
- def `write_certfr_cve` (self, certfr, cve)
Ecrit en BDD un binôme CERTFR/CVE calcul le CRC automatiquement.
- def `write_cve_tmp` (self, moncve)
Ecrit en BDD un CVE.
- def `write_cpe_tmp` (self, moncpe)
Ecrit en BDD un cpe.
- def `flush_tmp` (self)
Tranfère les tables tmp vers les mains.
- def `flush_url` (self)
Cherche tous les CERTFR mis à jour par les URL.
- def `get_all_new_certfr` (self)
Renvoie tous les bulletins mis à jour.
- def `get_certfr` (self, nom)
lit un bulletin en BDD
- def `get_all_cve_certfr` (self, certfr)
renvoie tous les CVE d'un bulletin
- def `get_max_lg_uri_cpe` (self, certfr)
Taille max des uri23 d'un bulletin.
- def `get_all_cpe_certfr` (self, certfr)
renvoie tous les CPE d'un bulletin
- def `get_all_cpe_uri` (self, uri)
renvoie tous les cpe pour un uri23
- def `get_orphan_by_obj` (self, obj)
Les bulletins par obj sans CVE.
- def `get_all_certfr_by_cve` (self)
Les bulletins par CVE.
- def `get_all_orphan` (self)
Bulletin par obj sans CVE.
- def `get_all_cve_orphan` (self)
Les CVE non présentes sur le NIST.
- def `load_mogs` (self)
Charge en BDD les couples CERTFR;CVE trouvés manuellement.

Attributs publics

- `Localdb`
le fichier MaBdd
- `moncur`
le curseur dans la Bdd

8.6.1 Description détaillée

Manipulation de la BDD.

Python help

Class pour interagir avec la BDD en sqlite3

8.6.2 Documentation des constructeurs et destructeur

8.6.2.1 `__init__()`

```
def RIA_sql.C_sql.__init__ (
    self )
```

constructors

Python help

le constructor
on utilise PRAGMA pour optimiser les écritures
ouvre le fichier RIA.db
Crée les tables de travail

8.6.3 Documentation des fonctions membres

8.6.3.1 `clean_new()`

```
def RIA_sql.C_sql.clean_new (
    self )
```

Met New à "0".

Python help

Met à 0 le champ New pour CERTFR,CVE et CVE_cpe

8.6.3.2 `clean_tmp()`

```
def RIA_sql.C_sql.clean_tmp (
    self )
```

efface les tables temporaires

Python help

Efface les tables temporaires CERTFR_tmp,CVE_tmp et CVE_cpe_tmp

8.6.3.3 close_db()

```
def RIA_sql.C_sql.close_db (
    self )
```

fermeture

Python help

fonction de sauvegarde et fermeture

8.6.3.4 flush_tmp()

```
def RIA_sql.C_sql.flush_tmp (
    self )
```

Tranfère les tables tmp vers les mains.

Python help

Transfère les données de
CERTFR_tmp vers CERTFR
CVE_tmp vers CVE
CVE_cpe_tmp vers CVE_cpe
gestion des doublons :
1 Hkey déjà présente
2 suppression des anciens enregistrements table officielle
3 copie des données restantes
4 efface les tables temporaires

8.6.3.5 flush_url()

```
def RIA_sql.C_sql.flush_url (
    self )
```

Cherche tous les CERTFR mis à jour par les URL.

Python help

Mise à jour du champ New des CERTFR par rapport au wrapper URL
1 URL_info(wrapper URL) vers CERTFR_Url (wrapper CERTFR)
2 CERTFR_Url vers CERTFR
3 URL_cve (wrapper URL) vers CVE (wrapper NIST CVE)
4 CVE vers CERTFR

8.6.3.6 `get_all_certfr_by_cve()`

```
def RIA_sql.C_sql.get_all_certfr_by_cve (
    self )
```

Les bulletins par CVE.

Renvoie

liste ou []

Python help

Renvoie une liste de tous les couples CVE/Bulletins

8.6.3.7 `get_all_cpe_certfr()`

```
def RIA_sql.C_sql.get_all_cpe_certfr (
    self,
    certfr )
```

renvoie tous les CPE d'un bulletin

Paramètres

<i>certfr</i>	nom du bulletin
---------------	-----------------

Renvoie

liste de C_cpe ou None

Python help

Renvoie une liste de C_cpe pour pour un bulletin
certfr est une string

8.6.3.8 `get_all_cpe_uri()`

```
def RIA_sql.C_sql.get_all_cpe_uri (
    self,
    uri )
```

renvoie tous les cpe pour un uri23

Paramètres

<i>uri</i>	une partie d'uri à chercher
------------	-----------------------------

Renvoie

liste de C_cpe ou None

Python help

```
Renvoie tous les cpe pour un uri23
recherche sql like %uri%
uri est une String
```

8.6.3.9 get_all_cve_certfr()

```
def RIA_sql.C_sql.get_all_cve_certfr (
    self,
    certfr )
```

renvoie tous les CVE d'un bulletin

Paramètres

<i>certfr</i>	nom du bulletin
---------------	-----------------

Renvoie

liste de C_cve ou une liste vide

Python help

```
Renvoie tous les CVE d'un bulletin ou une liste vide
certfr est un string
```

8.6.3.10 get_all_cve_orphan()

```
def RIA_sql.C_sql.get_all_cve_orphan (
    self )
```

Les CVE non présentes sur le NIST.

Renvoie

liste ou []

Python help

```
Renvoie une liste de toutes les CVE des bulletins non présentes sur le nist
Soit elles ne sont pas encore valides soit le CERTFR a mal formaté son bulletin
```

8.6.3.11 `get_all_new_certfr()`

```
def RIA_sql.C_sql.get_all_new_certfr (
    self )
```

Renvoie tous les bulletins mis à jour.

Renvoie

liste

Python help

renvoie une liste de tous les noms de bulletins avec New=1

8.6.3.12 `get_all_orphan()`

```
def RIA_sql.C_sql.get_all_orphan (
    self )
```

Bulletin par obj sans CVE.

Renvoie

liste ou []

Python help

Renvoie une liste de tous les bulletins et Objets sans CVE

8.6.3.13 `get_certfr()`

```
def RIA_sql.C_sql.get_certfr (
    self,
    nom )
```

lit un bulletin en BDD

Paramètres

<i>nom</i>	
------------	--

Renvoie

C_certfr

Python help

lit un bulletin dont le Nom ="nom"
nom est une String
renvoie un C_certfr (vide si pas trouvé en BDD)

8.6.3.14 get_Info_date()

```
def RIA_sql.C_sql.get_Info_date (
    self,
    quoi )
```

cherche une date dans la table information

Paramètres

<i>quoi</i>	ce que l'on recherche
-------------	-----------------------

A faire changer le nom des champs en plus explicite

Python help

Renvoie la date ou '' pour la recherche 'quoi' dans la table Info

8.6.3.15 get_max_lg_uri_cpe()

```
def RIA_sql.C_sql.get_max_lg_uri_cpe (
    self,
    certfr )
```

Taille max des uri23 d'un bulletin.

Paramètres

<i>certfr</i>	nom du bulletin
---------------	-----------------

Renvoie

la taille (nb de caractères)

A faire gérer les erreurs

Python help

Donne la taille max des uri23 pour un bulletin
certfr est une string

8.6.3.16 `get_orphan_by_obj()`

```
def RIA_sql.C_sql.get_orphan_by_obj (
    self,
    obj )
```

Les bulletins par obj sans CVE.

Paramètres

<i>obj</i>	chaîne à chercher
------------	-------------------

Renvoie

liste ou []

Python help

Renvoie une liste des noms de bulletins et objets sans CVE
obj chaîne à chercher dans les objets des bulletins
utilise la fonction SQL LIKE

8.6.3.17 `get_sc()`

```
def RIA_sql.C_sql.get_sc (
    self,
    script )
```

Exécute un script sqlite.

Paramètres

<i>script</i>	le sql
---------------	--------

Renvoie

une liste

Python help

Exécute le "script" SQL et renvoie une liste de tous les résultats

8.6.3.18 `load_mogs()`

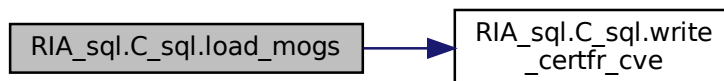
```
def RIA_sql.C_sql.load_mogs (
    self )
```

Charge en BDD les couples CERTFR;CVE trouvés manuellement.

Python help

Charge en BDD les couples CERTFR;CVE trouvés manuellement
les informations sont dans 'RIA_mogs.txt'
1 ligne par CERTFR;CVE

Voici le graphe d'appel pour cette fonction :



8.6.3.19 save_db()

```
def RIA_sql.C_sql.save_db (
    self )
```

Sauvegarde.

Python help

fonction de sauvegarde des transactions en cours

8.6.3.20 set_Info_date()

```
def RIA_sql.C_sql.set_Info_date (
    self,
    quoi,
    date )
```

écrit dans la table information.

Paramètres

<i>quoi</i>	le champ
<i>date</i>	la date

Python help

Sauvegarde la date pour la recherche 'quoi' dans la table Info

8.6.3.21 write_certfr_cve()

```
def RIA_sql.C_sql.write_certfr_cve (
    self,
    certfr,
    cve )
```

Ecrit en BDD un binôme CERTFR/CVE calcul le CRC automatiquement.

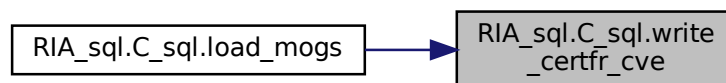
Paramètres

<i>certfr</i>	nom du bulletin
<i>cve</i>	nom du cve

Python help

Ecrit en BDD une CVE dans la table CERTFR_tmp
Et calcule la Hkey UNIQUE

Voici le graphe des appelants de cette fonction :



8.6.3.22 write_certfr_tmp()

```
def RIA_sql.C_sql.write_certfr_tmp (
    self,
    monbul )
```

Ecrit en BDD un bulletin.

Paramètres

<i>monbul</i>	un C_certfr
---------------	-------------

Python help

Ecrit dans la table CERTFR_tmp un bulletin
Ecrit les liens dans la table CERTFR_Url
monbul est un C_certfr

8.6.3.23 write_cpe_tmp()

```
def RIA_sql.C_sql.write_cpe_tmp (
    self,
    moncpe )
```

Ecrit en BDD un cpe.

Paramètres

<i>moncpe</i>	un C_cpe
---------------	----------

Python help

Ecrit en BDD un cpe dans la table CVE_cpe_tmp
moncpe est un C_cpe

8.6.3.24 write_cve_tmp()

```
def RIA_sql.C_sql.write_cve_tmp (
    self,
    moncve )
```

Ecrit en BDD un CVE.

Paramètres

<i>moncve</i>	un C_cve
---------------	----------

Python help

Ecrit en BDD un CVE dans la table CVE_tmp
moncve est un C_cve

8.6.3.25 write_sc()

```
def RIA_sql.C_sql.write_sc (
    self,
    script )
```

Exécute un script sqlite sans retour.

Paramètres

<i>script</i>	le sql
---------------	--------

Python help

Exécute le "script" SQL sans retour

8.6.4 Documentation des données membres

8.6.4.1 Localdb

`RIA_sql.C_sql.Localdb`

le fichier MaBdd

8.6.4.2 moncur

`RIA_sql.C_sql.moncur`

le curseur dans la Bdd

La documentation de cette classe a été générée à partir du fichier suivant :
— [package/RIA_sql.py](#)

8.7 Référence de la classe RIA_wrapper.C_wrapper

Class pour le Wrapper.

Graphe de collaboration de RIA_wrapper.C_wrapper:

RIA_wrapper.C_wrapper
+ MaBdd + mylog
+ __init__() + Reset_New() + Flush_cve() + write_url_cve() + Write_wrapper_info() + Read_wrapper_info() + Url_down_file() + Url_is_updated() + check_regex() + Download_Certfr() et 10 de plus...

Fonctions membres publiques

- def `__init__` (self, `MaBdd`)
constructors
- def `Reset_New` (self)
remet a 0 le champ New
- def `Flush_cve` (self)
Sauvegarde tous les couples Url/CVE trouvés.
- def `write_url_cve` (self, info)
Sauvegarde dans URL_ck.
- def `Write_wrapper_info` (self, info)
Sauvegarde en BDD un objet `C_wrapper_info`.
- def `Read_wrapper_info` (self, Champ, value, strict, New)
Lit en BDD des objet `C_wrapper_info`.
- def `Url_down_file` (self, info)
Télécharge un fichiers si plus récent.
- def `Url_is_updated` (self, info)
Verifie si le header de la page distante est plus recent.
- def `check_regex` (self, info)
Parse une url en regex.
- def `Download_Certfr` (self, EndDate)
Télécharge les Tar CERTFR si plus récent.
- def `Download_CVE` (self)
Télécharge les ZIP CVE NIST si plus récent.
- def `Load_ZIP_cve` (self, file)
extrait les info CVE d'un zip
- def `Search_re` (self, regex, obj)
Recherche regex pour Load_TAR_certfr.
- def `Load_TAR_certfr` (self, file)
extrait les info CERTFR d'un TAR
- def `Check_Gitlab` (self)
Verifie les release de la page about.gitlab.com.
- def `Check_Ubuntu` (self)
Verifie les release de la page <https://usn.ubuntu.com/months/>.
- def `Check_Kaspersky` (self)
Verifie Kaspersky.
- def `Check_Xen` (self)
Verifie Xen.
- def `Check_Redhat` (self)
Verifie Redhat.
- def `Check_ALL_Wapper_Update` (self, date)
Vérifie tous les editeurs en une seule fonction.

Attributs publics

- `MaBdd`
la Bdd via C_sql
- `mylog`

8.7.1 Description détaillée

Class pour le Wrapper.

Python help

```
Class pour le Data mining Internet
```

8.7.2 Documentation des constructeurs et destructeur

8.7.2.1 __init__()

```
def RIA_wrapper.C_wrapper.__init__ (
    self,
    MaBdd )
```

constructors

Paramètres

<i>MaBdd</i>	<i>C_sql</i>
--------------	--------------

Python help

le constructor
 MaBdd est un objet C_sql déjà ouvert
 On ajoute les tables SQL spécifiques
 info : Url est UNIQUE pour gérer les conflits d'INSERT

8.7.3 Documentation des fonctions membres

8.7.3.1 Check_ALL_Wapper_Update()

```
def RIA_wrapper.C_wrapper.Check_ALL_Wapper_Update (
    self,
    date )
```

Vérifie tous les editeurs en une seule fonction.

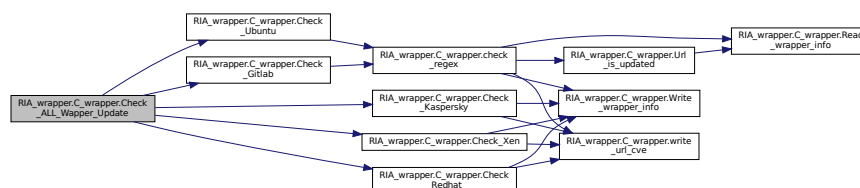
Paramètres

<i>date</i>	la date "YYYYMMDD" a verifier
-------------	-------------------------------

Python help

lance tous les wrapper en une seule fonction
 cela evite de devoir modifier RIA.py si l'on rajoute un nouveau.

Voici le graphe d'appel pour cette fonction :



8.7.3.2 Check_Gitlab()

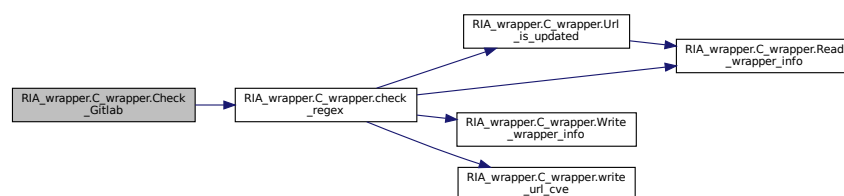
```
def RIA_wrapper.C_wrapper.Check_Gitlab (
    self )
```

Verifie les release de la page about.gitlab.com.

Python help

Verifie les release de la page about.gitlab.com

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.7.3.3 Check_Kaspersky()

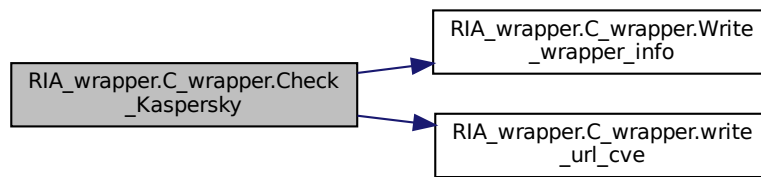
```
def RIA_wrapper.C_wrapper.Check_Kaspersky (
    self )
```

Verifie Kaspersky.

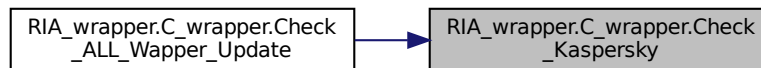
Python help

Vérifie les mise à jour kaspersky
On utilise BeautifulSoup pour la page aspx

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.7.3.4 Check_Redhat()

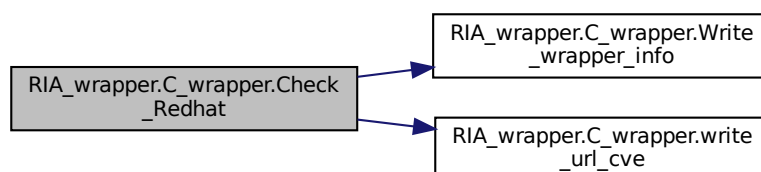
```
def RIA_wrapper.C_wrapper.Check_Redhat (
    self )
```

Verifie Redhat.

Python help

Verifie les CVRFs de RedHat via API

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.7.3.5 check_regex()

```
def RIA_wrapper.C_wrapper.check_regex (
    self,
    info )
```

Parse une url en regex.

Paramètres

<i>info</i>	Un C_wrapper_info
-------------	-----------------------------------

Renvoie

Boolean

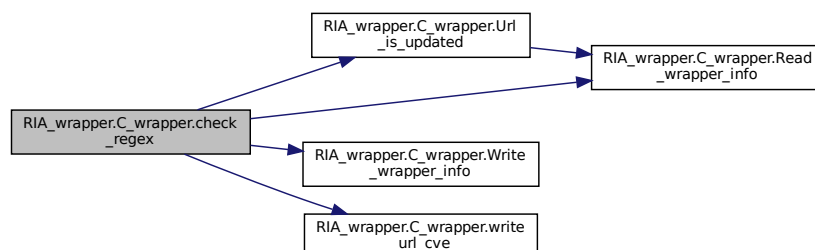
Python help

check_regex permet de parcourir une page unique avec des liens
 info doit être un objet C_wrapper_info

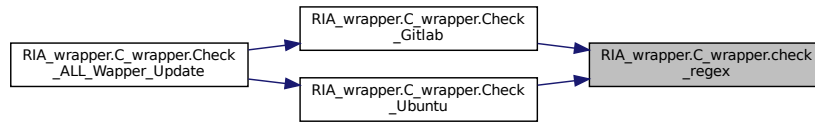
- Lit la page Url
- Cherche chaque sous-page avec le Regex
- Parse les sous-pages pour les CVE

revoie True sinon False si erreur

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.7.3.6 Check_Ubuntu()

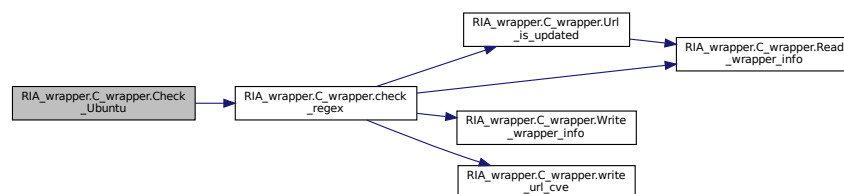
```
def RIA_wrapper.C_wrapper.Check_Ubuntu (
    self )
```

Verifie les release de la page <https://usn.ubuntu.com/months/>.

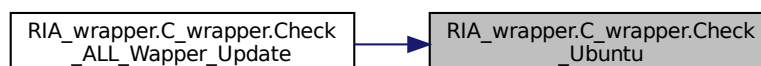
Python help

Verifie les release de la page <https://usn.ubuntu.com/months/>

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.7.3.7 Check_Xen()

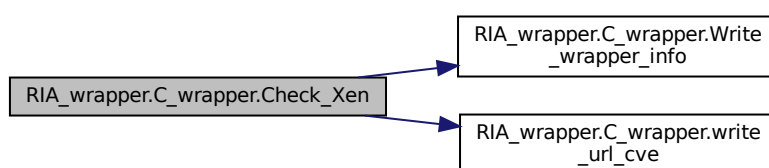
```
def RIA_wrapper.C_wrapper.Check_Xen (
    self )
```

Verifie Xen.

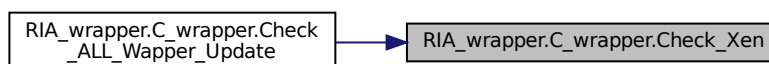
Python help

Verifie le JSON de Xen

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.7.3.8 Download_Certfr()

```
def RIA_wrapper.C_wrapper.Download_Certfr (
    self,
    EndDate )
```

Télécharge les Tar CERTFR si plus récent.

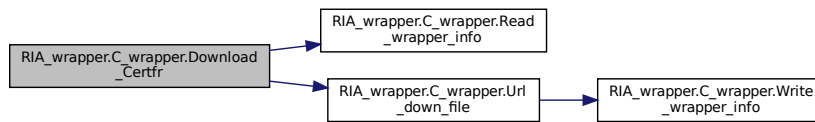
Paramètres

<i>EndDate</i>	l'année de fin int(YYYY)
----------------	--------------------------

Python help

Télécharge les fichiers année.tar du CERTFR
 EndDate est l'année en cours

Voici le graphe d'appel pour cette fonction :



8.7.3.9 Download_CVE()

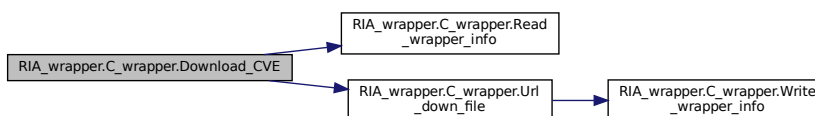
```
def RIA_wrapper.C_wrapper.Download_CVE (
    self )
```

Télécharge les ZIP CVE NIST si plus récent.

Python help

Télécharge les fichiers nvdcve-1.1-année.zip du NIST

Voici le graphe d'appel pour cette fonction :



8.7.3.10 Flush_cve()

```
def RIA_wrapper.C_wrapper.Flush_cve (
    self )
```

Sauvegarde tous les couples Url/CVE trouvés.

Python help

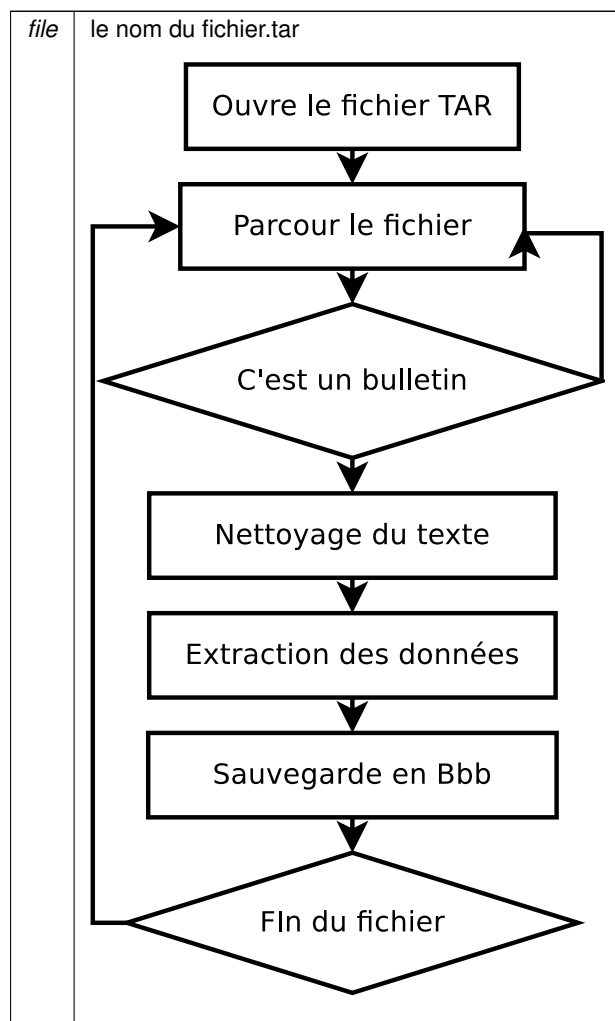
Transfert de la table TMP vers la table normale
 Selectionne tous les nom de bulletin et CVE où l'URL est commune aux deux tables
 Ajoute ensuite ces informations dans la liste officielle via write_certfr_cve

8.7.3.11 Load_TAR_certfr()

```
def RIA_wrapper.C_wrapper.Load_TAR_certfr (
    self,
    file )
```

extrait les info CERTFR d'un TAR

Paramètres

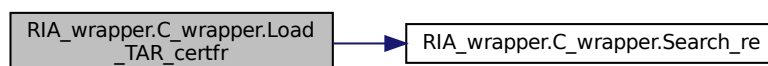


A faire nettoyer le code

Python help

```
Extrait toutes les informations d'un tar CERTFR  
file est avec son extension "nom_du_fichier.tar"
```

Voici le graphe d'appel pour cette fonction :

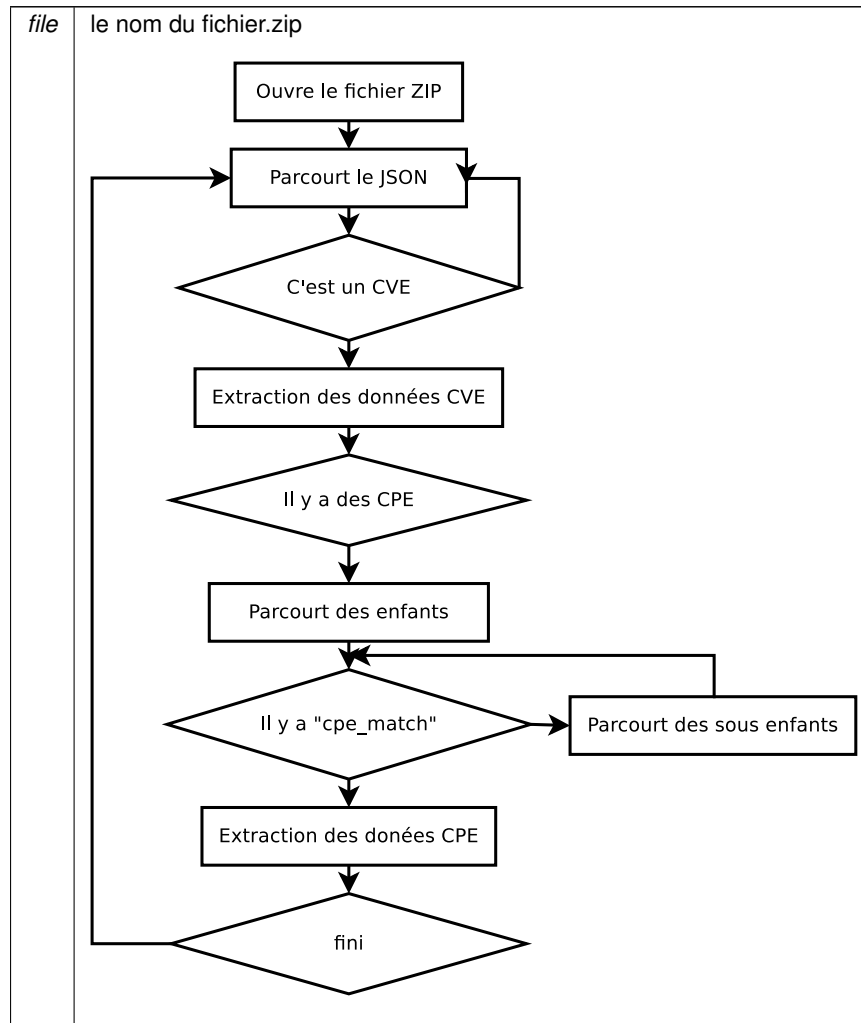


8.7.3.12 Load_ZIP_cve()

```
def RIA_wrapper.C_wrapper.Load_ZIP_cve (
    self,
    file )
```

extrait les info CVE d'un zip

Paramètres



A faire nettoyer le code

Python help

Extrait toutes les informations d'un zip CVE nist
file est avec son extension "nom_du_fichier.zip"

8.7.3.13 Read_wrapper_info()

```
def RIA_wrapper.C_wrapper.Read_wrapper_info (
    self,
    Champ,
    value,
    strict,
    New )
```

Lit en BDD des objet [C_wrapper_info](#).

Paramètres

<i>Champ</i>	le champ a chercher
<i>value</i>	la valeur a chercher
<i>strict</i>	pour recherche sql True: = sinon like "%%"
<i>New</i>	True ajoute " AND New=1;" a la recherche sql

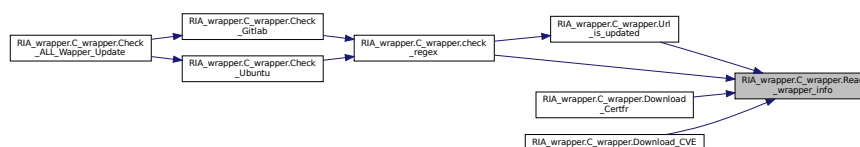
Renvoie

Une liste de [C_wrapper_info](#)

Python help

Lit une liste d'objet C_wrapper_info depuis la Bdd
 Strict(Boolean) permet de choisir entre une recherche '=' ou 'like'
 New(Boolean) permet de choisir que les New=1 ou pas

Voici le graphe des appelants de cette fonction :



8.7.3.14 Reset_New()

```
def RIA_wrapper.C_wrapper.Reset_New (
    self )
```

remet a 0 le champ New

Python help

Met a la valeur 0 le champ New en BDD

8.7.3.15 Search_re()

```
def RIA_wrapper.C_wrapper.Search_re (
    self,
    regex,
    obj )
```

Recherche regex pour Load_TAR_certfr.

Paramètres

<i>regex</i>	la regex
<i>obj</i>	la chaine a chercher

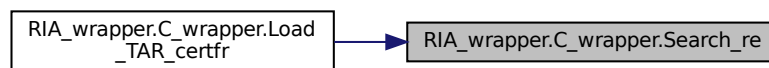
Renvoie

la string ou "

Python help

cherche la regex avec un group "()" dans obj
revoie dans tous les cas un String

Voici le graphe des appelants de cette fonction :



8.7.3.16 Url_down_file()

```
def RIA_wrapper.C_wrapper.Url_down_file (
    self,
    info )
```

Télécharge un fichiers si plus récent.

Paramètres

<i>info</i>	C_wrapper_info
-------------	--------------------------------

Python help

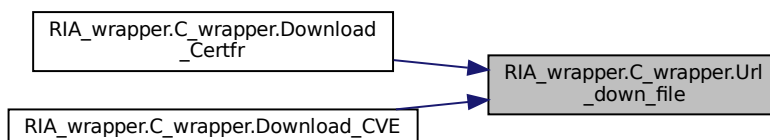
Télécharge un fichier s'il est plus récent

- vérifie la date du header vis a vis de la Bdd
- télécharge le fichiers
- met à jour la BDD

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.7.3.17 Url_is_updated()

```
def RIA_wrapper.C_wrapper.Url_is_updated (
    self,
    info )
```

Vérifie si le header de la page distante est plus récent.

Paramètres

<i>info</i>	le <code>C_wrapper_info</code> à vérifier
-------------	---

Renvoie

Boolean

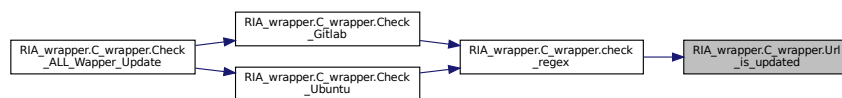
Python help

vérifie si le 'Last-Modified' est différent de la Bdd.
Si l'URL n'est pas en BDD renvoie True

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.7.3.18 write_url_cve()

```
def RIA_wrapper.C_wrapper.write_url_cve (
    self,
    info )
```

Sauvegarde dans URL_ck.

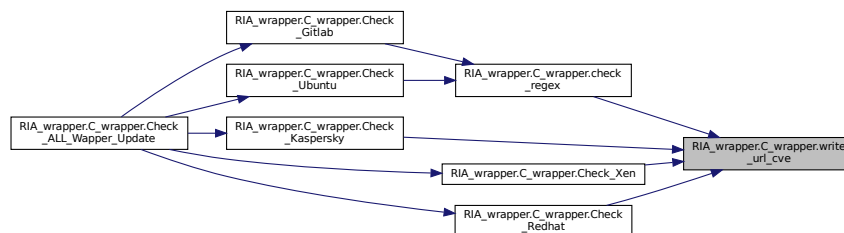
Paramètres

<i>info</i>	liste
-------------	-------

Python help

Insert en BDD
info est une liste [nom du bulletin,nom du CVE, date]

Voici le graphe des appelants de cette fonction :



8.7.3.19 Write_wrapper_info()

```
def RIA_wrapper.C_wrapper.Write_wrapper_info (
    self,
    info )
```

Sauvegarde en BDD un objet [C_wrapper_info](#).

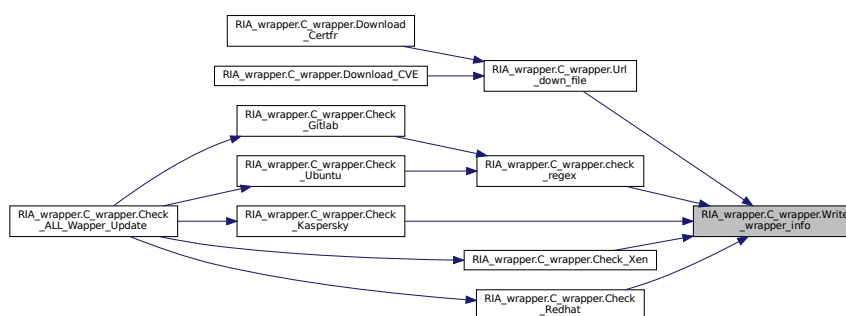
Paramètres

<i>info</i>	C_wrapper_info
-------------	--------------------------------

Python help

Insert en BDD
info est un C_wrapper_info

Voici le graphe des appelants de cette fonction :



8.7.4 Documentation des données membres

8.7.4.1 MaBdd

`RIA_wrapper.C_wrapper.MaBdd`

la Bdd via C_sql

8.7.4.2 mylog

`RIA_wrapper.C_wrapper.mylog`

La documentation de cette classe a été générée à partir du fichier suivant :

— [package/RIA_wrapper.py](#)

8.8 Référence de la classe RIA_wrapper.C_wrapper_info

Class Objet info Wrapper.

Graphe de collaboration de RIA_wrapper.C_wrapper_info:

RIA_wrapper.C_wrapper_info
<ul style="list-style-type: none"> + Url + Fichier + Taille + Rep + Date + Regex + S_Url + Module + New
+ __init__()

Fonctions membres publiques

- def `__init__` (self)
le constructor

Attributs publics

- `Url`
L'url.
- `Fichier`
Le fichier sert pour les CVE et CERTFR.
- `Taille`
La taille du fichier sans emploi maintenant.
- `Rep`
Le sous répertoire de sauvegarde du fichier.
- `Date`
Date de référence.
- `Regex`
la Regex de recherche de lien dans la page
- `S_Url`
le prefix a ajouter aux liens trouvés
- `Module`
Quelle fonction fait la recherche.
- `New`
Boolean Sqlite si nouveau.

8.8.1 Description détaillée

Class Objet info Wrapper.

Python help

Objet pour manipuler les informations sans passer par des listes

8.8.2 Documentation des constructeurs et destructeur

8.8.2.1 __init__()

```
def RIA_wrapper.C_wrapper_info.__init__ (
    self )
```

le constructor

Python help

constructor

8.8.3 Documentation des données membres

8.8.3.1 Date

RIA_wrapper.C_wrapper_info.Date

Date de référence.

8.8.3.2 Fichier

RIA_wrapper.C_wrapper_info.Fichier

Le fichier sert pour les CVE et CERTFR.

8.8.3.3 Module

RIA_wrapper.C_wrapper_info.Module

Quelle fonction fait la recherche.

8.8.3.4 New

RIA_wrapper.C_wrapper_info.New

Boolean Sqlite si nouveau.

8.8.3.5 Regex

`RIA_wrapper.C_wrapper_info.Regex`

la Regex de recherche de lien dans la page

8.8.3.6 Rep

`RIA_wrapper.C_wrapper_info.Rep`

Le sous répertoire de sauvegarde du fichier.

8.8.3.7 S_Url

`RIA_wrapper.C_wrapper_info.S_Url`

le prefix a ajouter aux liens trouvés

8.8.3.8 Taille

`RIA_wrapper.C_wrapper_info.Taille`

La taille du fichier sans emploi maintenant.

8.8.3.9 Url

`RIA_wrapper.C_wrapper_info.Url`

L'url.

La documentation de cette classe a été générée à partir du fichier suivant :

— package/[RIA_wrapper.py](#)

Chapitre 9

Documentation des fichiers

9.1 Référence du fichier package/RIA_class.py

Gestion des CERTFR,CVE et CPE.

Classes

- class [RIA_class.C_certfr](#)
Manipulation des CERTFR.
- class [RIA_class.C_cve](#)
Manipulation des CVE.
- class [RIA_class.C_cpe](#)
Manipulation CPE.

Espaces de nommage

- [RIA_class](#)

9.1.1 Description détaillée

Gestion des CERTFR,CVE et CPE.

Auteur

Frack113

Date

07/04/2020 Class pour les CERTFR,CVE et CPE Objet pour manipuler les bulletins,CVE ou CPE sans passer par le sql

9.2 Référence du fichier package/RIA_mskb.py

La gestion de l'API Microsoft.

Classes

- class [RIA_mskb.C_mskb](#)
Gestion API Microsoft.

Espaces de nommage

- [RIA_mskb](#)

9.2.1 Description détaillée

La gestion de l'API Microsoft.

Auteur

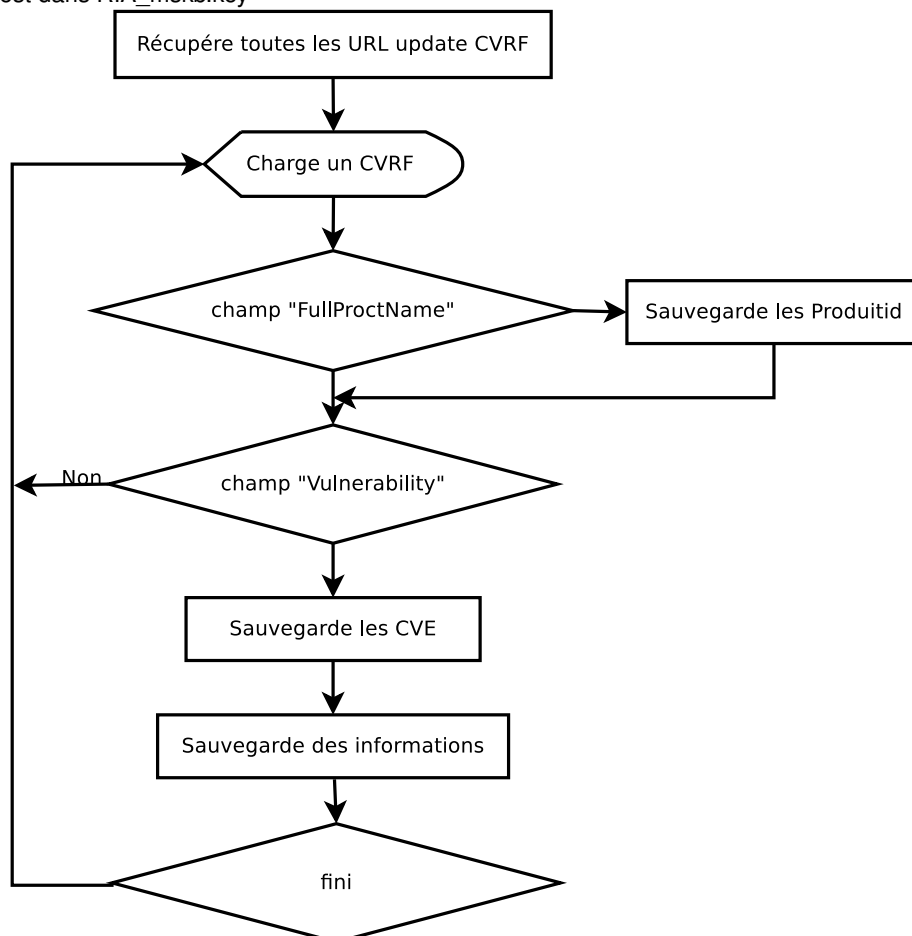
Frack113

Date

07/04/2020 Recherche de CVE et KB via l'API Microsoft

Avertissement

la clé API est dans RIA_mskb.key



A faire optimiser le code

peut être une "class" Wsus avec le cab <http://go.microsoft.com/fwlink/?linkid=74689>

9.3 Référence du fichier package/RIA_out.py

Gestion des sorties.

Classes

- class [RIA_out.C_out](#)
Gestion des sorties.

Espaces de nommage

- [RIA_out](#)

9.3.1 Description détaillée

Gestion des sorties.

Auteur

Frack113

Date

09/04/2020 Class pour les sorties fichier

9.4 Référence du fichier package/RIA_sql.py

La gestion de la base de données.

Classes

- class [RIA_sql.C_sql](#)
Manipulation de la BDD.

Espaces de nommage

- [RIA_sql](#)

9.4.1 Description détaillée

La gestion de la base de données.

Auteur

Frack113

Date

01/04/2020 Class pour les interactions avec la Bdd

Python help

9.5 Référence du fichier package/RIA_wrapper.py

La gestion des wrappers Internet.

Classes

- class [RIA_wrapper.C_wrapper_info](#)
Class Objet info Wrapper.
- class [RIA_wrapper.C_wrapper](#)
Class pour le Wrapper.

Espaces de nommage

- [RIA_wrapper](#)

9.5.1 Description détaillée

La gestion des wrappers Internet.

Auteur

Frack113

Date

08/04/2020 module pour les recherches Internet

9.6 Référence du fichier README.md

9.7 Référence du fichier RIA.py

Le coeur.

Espaces de nommage

- [RIA](#)

Fonctions

- def [RIA.credit](#) ()
Affiche simplement le crédit.
- def [RIA.mon_script](#) ()
Core du script.

9.7.1 Description détaillée

Le coeur.

Auteur

Frack113

Date

07/04/2020 Recherche d'Informations Automatisées

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

9.8 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

- `__init__`
 - `RIA_class.C_certfr`, [18](#)
 - `RIA_class.C_cpe`, [23](#)
 - `RIA_class.C_cve`, [27](#)
 - `RIA_mskb.C_mskb`, [31](#)
 - `RIA_out.C_out`, [38](#)
 - `RIA_sql.C_sql`, [44](#)
 - `RIA_wrapper.C_wrapper`, [55](#)
 - `RIA_wrapper.C_wrapper_info`, [71](#)
- `api_key`
 - `RIA_mskb.C_mskb`, [36](#)
- `api_type`
 - `RIA_mskb.C_mskb`, [36](#)
- `api_url`
 - `RIA_mskb.C_mskb`, [36](#)
- `CERT_to_STR`
 - `RIA_out.C_out`, [38](#)
- `Check_ALL_Wapper_Update`
 - `RIA_wrapper.C_wrapper`, [56](#)
- `Check_Gitlab`
 - `RIA_wrapper.C_wrapper`, [56](#)
- `Check_Kaspersky`
 - `RIA_wrapper.C_wrapper`, [57](#)
- `Check_Mskb_Update`
 - `RIA_mskb.C_mskb`, [31](#)
- `Check_Redhat`
 - `RIA_wrapper.C_wrapper`, [58](#)
- `check_regex`
 - `RIA_wrapper.C_wrapper`, [59](#)
- `Check_Ubuntu`
 - `RIA_wrapper.C_wrapper`, [60](#)
- `Check_Xen`
 - `RIA_wrapper.C_wrapper`, [60](#)
- `clean_new`
 - `RIA_sql.C_sql`, [44](#)
- `clean_tmp`
 - `RIA_sql.C_sql`, [44](#)
- `close_db`
 - `RIA_sql.C_sql`, [44](#)
- `conf`
 - `RIA_class.C_cpe`, [24](#)
- `cpe23Uri`
 - `RIA_class.C_cpe`, [24](#)
- `crc`
 - `RIA_class.C_certfr`, [20](#)
 - `RIA_class.C_cpe`, [24](#)
 - `RIA_class.C_cve`, [28](#)
- `credit`
 - `RIA`, [13](#)
- `cve`
 - `RIA_class.C_cpe`, [24](#)
- `cvssV2`
 - `RIA_class.C_cve`, [28](#)
- `cvssV2base`
 - `RIA_class.C_cve`, [28](#)
- `cvssV3`
 - `RIA_class.C_cve`, [29](#)
- `cvssV3base`
 - `RIA_class.C_cve`, [29](#)
- `Date`
 - `RIA_wrapper.C_wrapper_info`, [71](#)
- `dateOrigine`
 - `RIA_class.C_certfr`, [20](#)
 - `RIA_class.C_cve`, [29](#)
- `dateUpdate`
 - `RIA_class.C_certfr`, [21](#)
 - `RIA_class.C_cve`, [29](#)
- `decode_file`
 - `RIA_class.C_certfr`, [19](#)
- `decode_link`
 - `RIA_class.C_certfr`, [19](#)
- `Download_Certfr`
 - `RIA_wrapper.C_wrapper`, [61](#)
- `Download_CVE`
 - `RIA_wrapper.C_wrapper`, [62](#)
- `encode_file`
 - `RIA_class.C_certfr`, [19](#)
- `encode_link`
 - `RIA_class.C_certfr`, [19](#)
- `Export_certfr_json`
 - `RIA_out.C_out`, [39](#)
 - `test_json`, [15](#)
- `Fichier`
 - `RIA_wrapper.C_wrapper_info`, [71](#)
- `file`
 - `RIA_class.C_certfr`, [21](#)
- `Flush_cve`
 - `RIA_wrapper.C_wrapper`, [62](#)
- `flush_tmp`
 - `RIA_sql.C_sql`, [45](#)
- `flush_url`
 - `RIA_sql.C_sql`, [45](#)
- `get_all_certfr_by_cve`
 - `RIA_sql.C_sql`, [45](#)

- get_all_cpe_certfr
 - RIA_sql.C_sql, [46](#)
- get_all_cpe_uri
 - RIA_sql.C_sql, [46](#)
- get_all_cve_certfr
 - RIA_sql.C_sql, [47](#)
- get_all_cve_orphan
 - RIA_sql.C_sql, [47](#)
- get_all_new_certfr
 - RIA_sql.C_sql, [47](#)
- get_all_orphan
 - RIA_sql.C_sql, [48](#)
- get_certfr
 - RIA_sql.C_sql, [48](#)
- get_info_certfr
 - RIA_mskb.C_mskb, [32](#)
- get_Info_date
 - RIA_sql.C_sql, [49](#)
- get_max_lg_uri_cpe
 - RIA_sql.C_sql, [49](#)
- get_orphan_by_obj
 - RIA_sql.C_sql, [49](#)
- get_sc
 - RIA_sql.C_sql, [50](#)
- header
 - RIA_mskb.C_mskb, [36](#)
- html
 - test_json, [15](#)
- id
 - RIA_class.C_cpe, [25](#)
 - RIA_class.C_cve, [29](#)
- Ksoft
 - RIA_out.C_out, [41](#)
- link
 - RIA_class.C_certfr, [21](#)
- load_mogs
 - RIA_sql.C_sql, [50](#)
- Load_TAR_certfr
 - RIA_wrapper.C_wrapper, [62](#)
- Load_ZIP_cve
 - RIA_wrapper.C_wrapper, [63](#)
- Localdb
 - RIA_sql.C_sql, [54](#)
- MaBdd
 - RIA_mskb.C_mskb, [36](#)
 - RIA_out.C_out, [42](#)
 - RIA_wrapper.C_wrapper, [69](#)
 - test_json, [15](#)
- Module
 - RIA_wrapper.C_wrapper_info, [71](#)
- mon_script
 - RIA, [13](#)
- moncur
 - RIA_sql.C_sql, [54](#)
- MS_to_STR
 - RIA_out.C_out, [39](#)
- mylog
 - RIA_wrapper.C_wrapper, [69](#)
- New
 - RIA_class.C_certfr, [21](#)
 - RIA_class.C_cpe, [25](#)
 - RIA_class.C_cve, [29](#)
 - RIA_wrapper.C_wrapper_info, [71](#)
- nom
 - RIA_class.C_certfr, [21](#)
- obj
 - RIA_class.C_certfr, [21](#)
- opérateur
 - RIA_class.C_cpe, [25](#)
- package/RIA_class.py, [73](#)
- package/RIA_mskb.py, [73](#)
- package/RIA_out.py, [75](#)
- package/RIA_sql.py, [75](#)
- package/RIA_wrapper.py, [76](#)
- Read_wrapper_info
 - RIA_wrapper.C_wrapper, [64](#)
- README.md, [76](#)
- Regex
 - RIA_wrapper.C_wrapper_info, [71](#)
- Rep
 - RIA_wrapper.C_wrapper_info, [72](#)
- reset
 - RIA_class.C_certfr, [20](#)
 - RIA_class.C_cpe, [23](#)
 - RIA_class.C_cve, [28](#)
- reset_db
 - RIA_mskb.C_mskb, [33](#)
- Reset_New
 - RIA_wrapper.C_wrapper, [65](#)
- RIA, [13](#)
 - credit, [13](#)
 - mon_script, [13](#)
- RIA.py, [76](#)
- RIA_class, [14](#)
- RIA_class.C_certfr, [17](#)
 - __init__, [18](#)
 - crc, [20](#)
 - dateOrigine, [20](#)
 - dateUpdate, [21](#)
 - decode_file, [19](#)
 - decode_link, [19](#)
 - encode_file, [19](#)
 - encode_link, [19](#)
 - file, [21](#)
 - link, [21](#)
 - New, [21](#)
 - nom, [21](#)
 - obj, [21](#)
 - reset, [20](#)

- set_crc, 20
- RIA_class.C_cpe, 22
 - __init__, 23
 - conf, 24
 - cpe23Uri, 24
 - crc, 24
 - cve, 24
 - id, 25
 - New, 25
 - opérateur, 25
 - reset, 23
 - set_crc, 24
 - versionEndExcluding, 25
 - versionEndIncluding, 25
 - versionStartExcluding, 25
 - versionStartIncluding, 26
 - vulnerable, 26
- RIA_class.C_cve, 26
 - __init__, 27
 - crc, 28
 - cvssV2, 28
 - cvssV2base, 28
 - cvssV3, 29
 - cvssV3base, 29
 - dateOrigine, 29
 - dateUpdate, 29
 - id, 29
 - New, 29
 - reset, 28
 - set_crc, 28
- RIA_mskb, 14
- RIA_mskb.C_mskb, 30
 - __init__, 31
 - api_key, 36
 - api_type, 36
 - api_url, 36
 - Check_Mskb_Update, 31
 - get_info_certfr, 32
 - header, 36
 - MaBdd, 36
 - reset_db, 33
 - update_all_info, 33
 - update_all_url, 34
 - write_cve_kb, 34
 - write_product, 35
- RIA_out, 14
- RIA_out.C_out, 37
 - __init__, 38
 - CERT_to_STR, 38
 - Export_certfr_json, 39
 - Ksoft, 41
 - MaBdd, 42
 - MS_to_STR, 39
 - tab2_to_txt, 40
 - URI_to_FILE, 40
 - Write_CERTFR, 41
- RIA_sql, 15
- RIA_sql.C_sql, 42
 - __init__, 44
 - clean_new, 44
 - clean_tmp, 44
 - close_db, 44
 - flush_tmp, 45
 - flush_url, 45
 - get_all_certfr_by_cve, 45
 - get_all_cpe_certfr, 46
 - get_all_cpe_uri, 46
 - get_all_cve_certfr, 47
 - get_all_cve_orphan, 47
 - get_all_new_certfr, 47
 - get_all_orphan, 48
 - get_certfr, 48
 - get_Info_date, 49
 - get_max_lg_uri_cpe, 49
 - get_orphan_by_obj, 49
 - get_sc, 50
 - load_mogs, 50
 - Localdb, 54
 - moncur, 54
 - save_db, 51
 - set_Info_date, 51
 - write_certfr_cve, 51
 - write_certfr_tmp, 52
 - write_cpe_tmp, 52
 - write_cve_tmp, 53
 - write_sc, 53
- RIA_wrapper, 15
- RIA_wrapper.C_wrapper, 54
 - __init__, 55
 - Check_ALL_Wapper_Update, 56
 - Check_Gitlab, 56
 - Check_Kaspersky, 57
 - Check_Redhat, 58
 - check_regex, 59
 - Check_Ubuntu, 60
 - Check_Xen, 60
 - Download_Certfr, 61
 - Download_CVE, 62
 - Flush_cve, 62
 - Load_TAR_certfr, 62
 - Load_ZIP_cve, 63
 - MaBdd, 69
 - mylog, 69
 - Read_wrapper_info, 64
 - Reset_New, 65
 - Search_re, 65
 - Url_down_file, 66
 - Url_is_updated, 67
 - write_url_cve, 68
 - Write_wrapper_info, 68
- RIA_wrapper.C_wrapper_info, 70
 - __init__, 71
 - Date, 71
 - Fichier, 71
 - Module, 71
 - New, 71

- Regex, [71](#)
- Rep, [72](#)
- S_Url, [72](#)
- Taille, [72](#)
- Url, [72](#)
- S_Url
 - RIA_wrapper.C_wrapper_info, [72](#)
- save_db
 - RIA_sql.C_sql, [51](#)
- Search_re
 - RIA_wrapper.C_wrapper, [65](#)
- set_crc
 - RIA_class.C_certfr, [20](#)
 - RIA_class.C_cpe, [24](#)
 - RIA_class.C_cve, [28](#)
- set_Info_date
 - RIA_sql.C_sql, [51](#)
- tab2_to_txt
 - RIA_out.C_out, [40](#)
- Taille
 - RIA_wrapper.C_wrapper_info, [72](#)
- test_json, [15](#)
 - Export_certfr_json, [15](#)
 - html, [15](#)
 - MaBdd, [15](#)
- test_json.py, [77](#)
- update_all_info
 - RIA_mskb.C_mskb, [33](#)
- update_all_url
 - RIA_mskb.C_mskb, [34](#)
- URI_to_FILE
 - RIA_out.C_out, [40](#)
- Url
 - RIA_wrapper.C_wrapper_info, [72](#)
- Url_down_file
 - RIA_wrapper.C_wrapper, [66](#)
- Url_is_updated
 - RIA_wrapper.C_wrapper, [67](#)
- versionEndExcluding
 - RIA_class.C_cpe, [25](#)
- versionEndIncluding
 - RIA_class.C_cpe, [25](#)
- versionStartExcluding
 - RIA_class.C_cpe, [25](#)
- versionStartIncluding
 - RIA_class.C_cpe, [26](#)
- vulnerable
 - RIA_class.C_cpe, [26](#)
- Write_CERTFR
 - RIA_out.C_out, [41](#)
- write_certfr_cve
 - RIA_sql.C_sql, [51](#)
- write_certfr_tmp
 - RIA_sql.C_sql, [52](#)
- write_cpe_tmp
 - RIA_sql.C_sql, [52](#)
- write_cve_kb
 - RIA_mskb.C_mskb, [34](#)
- write_cve_tmp
 - RIA_sql.C_sql, [53](#)
- write_product
 - RIA_mskb.C_mskb, [35](#)
- write_sc
 - RIA_sql.C_sql, [53](#)
- write_url_cve
 - RIA_wrapper.C_wrapper, [68](#)
- Write_wrapper_info
 - RIA_wrapper.C_wrapper, [68](#)