

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 CERT_to_STR()	13
7.1.1.2 credit()	14
7.1.1.3 mon_script()	14
7.1.1.4 MS_to_STR()	15
7.1.1.5 URI_to_FILE()	15
7.1.1.6 Write_CERTFR()	16
7.2 Référence de l'espace de nommage RIA_class	17
7.3 Référence de l'espace de nommage RIA_mskb	17
7.4 Référence de l'espace de nommage RIA_sql	17
7.5 Référence de l'espace de nommage RIA_wrapper	17
8 Documentation des classes	19
8.1 Référence de la classe RIA_class.C_certfr	19
8.1.1 Description détaillée	20
8.1.2 Documentation des constructeurs et destructeur	20
8.1.2.1 __init__()	20
8.1.3 Documentation des fonctions membres	21
8.1.3.1 decode_file()	21
8.1.3.2 decode_link()	21
8.1.3.3 encode_file()	21
8.1.3.4 encode_link()	22
8.1.3.5 reset()	22
8.1.3.6 set_crc()	22
8.1.4 Documentation des données membres	22
8.1.4.1 crc	22

8.1.4.2 dateOrigine	23
8.1.4.3 dateUpdate	23
8.1.4.4 file	23
8.1.4.5 link	23
8.1.4.6 New	23
8.1.4.7 nom	23
8.1.4.8 obj	24
8.2 Référence de la classe RIA_class.C_cpe	24
8.2.1 Description détaillée	25
8.2.2 Documentation des constructeurs et destructeur	25
8.2.2.1 __init__()	25
8.2.3 Documentation des fonctions membres	25
8.2.3.1 reset()	26
8.2.3.2 set_crc()	26
8.2.4 Documentation des données membres	26
8.2.4.1 conf	26
8.2.4.2 cpe23Uri	26
8.2.4.3 crc	26
8.2.4.4 cve	27
8.2.4.5 id	27
8.2.4.6 New	27
8.2.4.7 operateur	27
8.2.4.8 versionEndExcluding	27
8.2.4.9 versionEndIncluding	27
8.2.4.10 versionStartExcluding	28
8.2.4.11 versionStartIncluding	28
8.2.4.12 vulnerable	28
8.3 Référence de la classe RIA_class.C_cve	28
8.3.1 Description détaillée	29
8.3.2 Documentation des constructeurs et destructeur	29
8.3.2.1 __init__()	29
8.3.3 Documentation des fonctions membres	30
8.3.3.1 reset()	30
8.3.3.2 set_crc()	30
8.3.4 Documentation des données membres	30
8.3.4.1 crc	30
8.3.4.2 cvssV2	30
8.3.4.3 cvssV2base	31
8.3.4.4 cvssV3	31
8.3.4.5 cvssV3base	31
8.3.4.6 dateOrigine	31
8.3.4.7 dateUpdate	31

8.3.4.8 id	31
8.3.4.9 New	32
8.4 Référence de la classe RIA_mskb.C_mskb	32
8.4.1 Description détaillée	33
8.4.2 Documentation des constructeurs et destructeur	33
8.4.2.1 __init__()	33
8.4.3 Documentation des fonctions membres	33
8.4.3.1 Check_Mskb_Update()	34
8.4.3.2 get_info_certfr()	34
8.4.3.3 reset_db()	35
8.4.3.4 update_all_info()	35
8.4.3.5 update_all_url()	36
8.4.3.6 write_cve_kb()	36
8.4.3.7 write_product()	37
8.4.4 Documentation des données membres	37
8.4.4.1 api_key	37
8.4.4.2 api_type	38
8.4.4.3 api_url	38
8.4.4.4 header	38
8.4.4.5 MaBdd	38
8.5 Référence de la classe RIA_sql.C_sql	38
8.5.1 Description détaillée	39
8.5.2 Documentation des constructeurs et destructeur	40
8.5.2.1 __init__()	40
8.5.3 Documentation des fonctions membres	40
8.5.3.1 clean_new()	40
8.5.3.2 clean_tmp()	40
8.5.3.3 close_db()	41
8.5.3.4 flush_tmp()	41
8.5.3.5 flush_url()	41
8.5.3.6 get_all_certfr_by_cve()	42
8.5.3.7 get_all_cpe_certfr()	42
8.5.3.8 get_all_cpe_uri()	42
8.5.3.9 get_all_cve_certfr()	43
8.5.3.10 get_all_cve_orphan()	43
8.5.3.11 get_all_new_certfr()	44
8.5.3.12 get_all_orphan()	44
8.5.3.13 get_certfr()	44
8.5.3.14 get_Info_date()	45
8.5.3.15 get_max_lg_uri_cpe()	45
8.5.3.16 get_orphan_by_obj()	46
8.5.3.17 get_sc()	46

8.5.3.18 load_mogs()	46
8.5.3.19 save_db()	47
8.5.3.20 set_Info_date()	47
8.5.3.21 write_certfr_cve()	48
8.5.3.22 write_certfr_tmp()	48
8.5.3.23 write_cpe_tmp()	49
8.5.3.24 write_cve_tmp()	49
8.5.3.25 write_sc()	49
8.5.4 Documentation des données membres	50
8.5.4.1 Localdb	50
8.5.4.2 moncur	50
8.6 Référence de la classe RIA_wrapper.C_wrapper	50
8.6.1 Description détaillée	51
8.6.2 Documentation des constructeurs et destructeur	51
8.6.2.1 __init__()	52
8.6.3 Documentation des fonctions membres	52
8.6.3.1 Check_ALL_Wapper_Update()	52
8.6.3.2 Check_Gitlab()	53
8.6.3.3 Check_Kaspersky()	53
8.6.3.4 Check_Redhat()	54
8.6.3.5 check_regex()	55
8.6.3.6 Check_Ubuntu()	56
8.6.3.7 Check_Xen()	57
8.6.3.8 Download_Certfr()	57
8.6.3.9 Download_CVE()	58
8.6.3.10 Flush_cve()	58
8.6.3.11 Load_TAR_certfr()	58
8.6.3.12 Load_ZIP_cve()	60
8.6.3.13 Read_wrapper_info()	61
8.6.3.14 Reset_New()	61
8.6.3.15 Search_re()	62
8.6.3.16 Url_down_file()	62
8.6.3.17 Url_is_updated()	63
8.6.3.18 write_url_cve()	64
8.6.3.19 Write_wrapper_info()	65
8.6.4 Documentation des données membres	65
8.6.4.1 MaBdd	65
8.7 Référence de la classe RIA_wrapper.C_wrapper_info	66
8.7.1 Description détaillée	66
8.7.2 Documentation des constructeurs et destructeur	67
8.7.2.1 __init__()	67
8.7.3 Documentation des données membres	67

8.7.3.1 Date	67
8.7.3.2 Fichier	67
8.7.3.3 Module	67
8.7.3.4 New	67
8.7.3.5 Regex	68
8.7.3.6 Rep	68
8.7.3.7 S_Url	68
8.7.3.8 Taille	68
8.7.3.9 Url	68
9 Documentation des fichiers	69
9.1 Référence du fichier README.md	69
9.2 Référence du fichier RIA.py	69
9.2.1 Description détaillée	69
9.3 Référence du fichier RIA_class.py	70
9.3.1 Description détaillée	70
9.4 Référence du fichier RIA_mskb.py	70
9.4.1 Description détaillée	71
9.5 Référence du fichier RIA_sql.py	71
9.5.1 Description détaillée	71
9.6 Référence du fichier RIA_wrapper.py	71
9.6.1 Description détaillée	72
Index	73

Chapitre 1

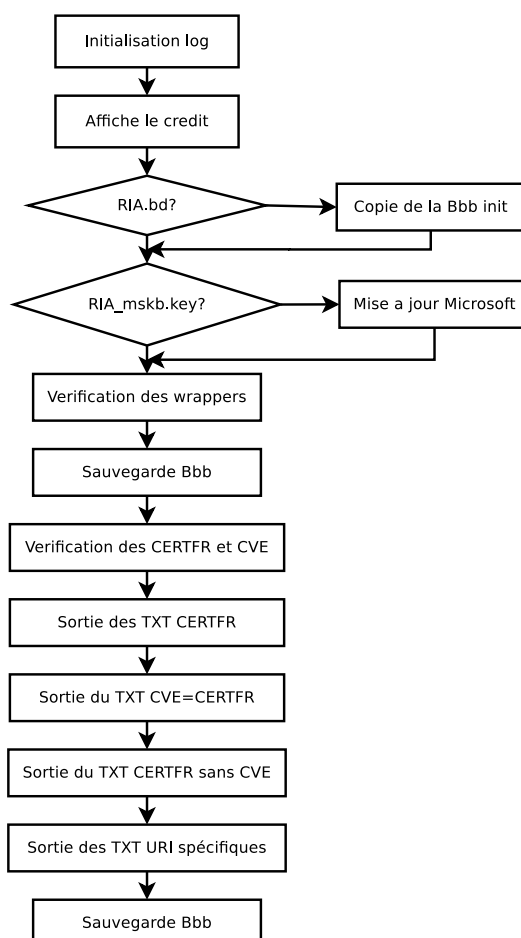
Page principale

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, creer une class...

Compléter le bulletin du CERTFR avec le plus d'informations possibles

Le code est donc pas propre, optimisée, voir clair.

Test sous FEDORA 31 , Python 3.7.6

Actuellement

- Téléchargement automatique des CERTFR et CVE
- Lecture des bulletin CERTFR depuis archive rar [1](#)
- Lecture des cve depuis les json [2](#)
- Gestion en SQLITE
- Liste des KB via **API** microsoft
- Wrapper site éditeur pour trouver les CVE
- Ajout de l'aide des **Mogs** (ajout manuel)
- mise a jour norme PEP 257 [3](#)

Todo

- nettoyer le code
- Ajouter au Wrapper
- ?..

Documentation

- Doxygen
- Graphviz
- Textlive pour LATEX to PDF

Références

Chapitre 3

Liste des choses à faire

Membre `RIA.mon_script ()`

simplifier les repetitions

Fichier `RIA.py`

utiliser les best practice Python

Fichier `RIA_mskb.py`

optimiser le code

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

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

changer le nom des champs plus clair

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	17
RIA_mskb	17
RIA_sql	17
RIA_wrapper	17

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	19
RIA_class.C_cpe	
Manipulation CPE	24
RIA_class.C_cve	
Manipulation des CVE	28
RIA_mskb.C_mskb	
Gestion API Microsoft	32
RIA_sql.C_sql	38
RIA_wrapper.C_wrapper	
Class pour le Wrapper	50
RIA_wrapper.C_wrapper_info	
Class Objet info Wrapper	66

Chapitre 6

Index des fichiers

6.1 Liste des fichiers

Liste de tous les fichiers avec une brève description :

RIA.py	Recherche d'Information Automatisée	69
RIA_class.py	Gestion des CERTFR,CVE et CPE	70
RIA_mskb.py	La gestion de l'API Microsoft	70
RIA_sql.py	La gestion de la base de données	71
RIA_wrapper.py	La gestion des wrapper Internet	71

Chapitre 7

Documentation des espaces de nommage

7.1 Référence de l'espace de nommage RIA

Fonctions

- def `credit` ()
Affiche simplement le credit.
- def `CERT_to_STR` (Nom, tab)
Une jolie sortie formater des info CERTFR.
- def `MS_to_STR` (Nom, tab)
Une jolie sortie formater des info Microsoft.
- def `Write_CERTFR` (nom, annee)
Ecrit dans un fichier text les informations du bulletin.
- def `URI_to_FILE` (Nom, uri)
Ecrit un fichiers avec bulletins et URI23 pour une recherche.
- def `mon_script` ()
Core du scripts.

7.1.1 Documentation des fonctions

7.1.1.1 `CERT_to_STR()`

```
def RIA.CERT_to_STR (  
    Nom,  
    tab )
```

Une jolie sortie formater des info CERTFR.

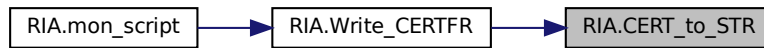
Paramètres

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

Renvoie

None

Voici le graphe des appelants de cette fonction :

**7.1.1.2 credit()**

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

Affiche simplement le credit.

Voici le graphe des appelants de cette fonction :

**7.1.1.3 mon_script()**

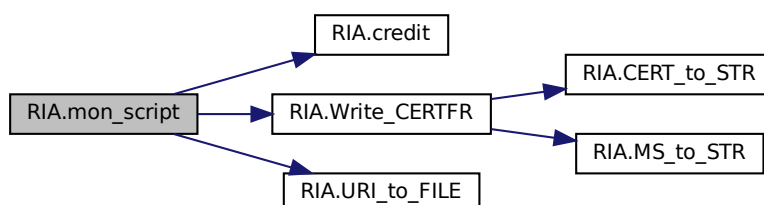
```
def RIA.mon_script ( )
```

Core du scripts.

le coeur du scripts

A faire simplifier les repetitions

Voici le graphe d'appel pour cette fonction :



7.1.1.4 MS_to_STR()

```
def RIA.MS_to_STR (
    Nom,
    tab )
```

Une jolie sortie formater des info Microsoft.

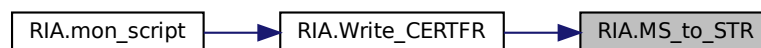
Paramètres

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

Renvoie

None

Voici le graphe des appelants de cette fonction :



7.1.1.5 URI_to_FILE()

```
def RIA.URI_to_FILE (
    Nom,
    uri )
```

Ecrit un fichiers avec bulletins et URI23 pour une recherche.

Paramètres

<i>Nom</i>	dans les objets et non du fichier de sortie
<i>uri</i>	chaîne à chercher dans les uri23

Voici le graphe des appelants de cette fonction :



7.1.1.6 Write_CERTFR()

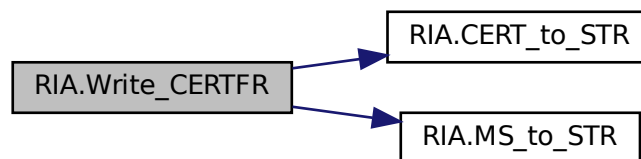
```
def RIA.Write_CERTFR (  
    nom,  
    annee )
```

Ecrit dans un fichier text les informations du bulletin.

Paramètres

<i>nom</i>	le nom du bulletin
<i>annee</i>	repertoire de sortie

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de 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_sql

Classes

- class [C_sql](#)

7.5 Référence de l'espace de nommage RIA_wrapper

Classes

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

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)

```

— Calcul le CRC pour la clé UNIQUE SQL.
— def decode_file (self)
    decode 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)
    decode les liens en base64

```

Attributs publics

```

— nom
    le nom du bulletin
— obj
    l'objet du bulletin
— dateOrigine
    Date de creation du bulletin.
— dateUpdate
    Date de modification du bulletin.
— New
    boolean 0 deja 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 repressente 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 )
```

decode 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 )
```

decode les liens en base64

Avertissement

surement 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

Encadage 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

surement 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 a l'état d'origine les variables

8.1.3.6 `set_crc()`

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

Calcul le CRC pour la clé UNIQUE SQL.

Python help

calcul 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 creation 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 deja 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

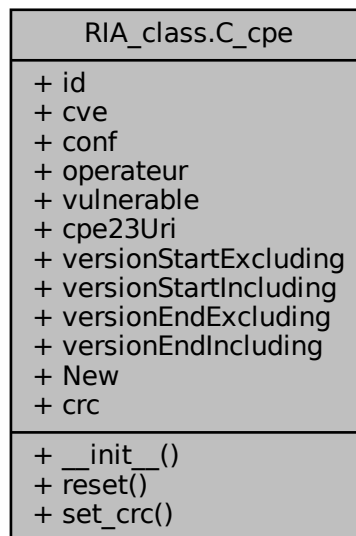
La documentation de cette classe a été générée à partir du fichier suivant :

— [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)`
Calcul 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 vulnerable ex Firefox sur windows firefox AND (false) Windows.
- `cpe23Uri`
l'URI en 2.3
- `versionStartExcluding`
Version de depart exclue.
- `versionStartIncluding`
Version de depart inclue.
- `versionEndExcluding`
Version de fin exclue.
- `versionEndIncluding`
Version de fin inclue.
- `New`
boolean 0 deja traité , 1 nouveau
- `crc`
Clé unique.

8.2.1 Description détaillée

Manipulation CPE.

Python help

Class qui represente 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 a l'état d'origine les variables

8.2.3.2 set_crc()

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

Calcul le CRC pour la clé UNIQUE SQL.

Python help

calcul 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 deja traité , 1 nouveau

8.2.4.7 operateur

`RIA_class.C_cpe.operateur`

Operateur 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 inclue.

8.2.4.10 versionStartExcluding

`RIA_class.C_cpe.versionStartExcluding`

Version de depart exclue.

8.2.4.11 versionStartIncluding

`RIA_class.C_cpe.versionStartIncluding`

Version de depart inclue.

8.2.4.12 vulnerable

`RIA_class.C_cpe.vulnerable`

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

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

— [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)
Calcul 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 creation du CVE.
- `dateUpdate`
La date derniere modification du CVE.
- `New`
boolean 0 deja traité , 1 nouveau
- `crc`
Clé unique.

8.3.1 Description détaillée

Manipulation des CVE.

Python help

```
Class qui repressente 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 a l'état d'origine les variables

8.3.3.2 set_crc()

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

Calcul le CRC pour la clé UNIQUE SQL.

Python help

calcul 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 creation du CVE.

8.3.4.7 `dateUpdate`

`RIA_class.C_cve.dateUpdate`

La date derniere 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 deja traité , 1 nouveau

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

— [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)
recupere 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)
recupere toutes les information
- def `get_info_certfr` (self, certfr)
Cherche les information pour un CERTFR.
- def `Check_Mskb_Update` (self, date)
Verifie la mise a 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 C_sql déjà initialisé
il faut une clé API sauvegardée dans RIA_mskb.key
crée automatiquement les tables SQL
```

8.4.3 Documentation des fonctions membres

8.4.3.1 Check_Mskb_Update()

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

Verifie la mise a jour.

Paramètres

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

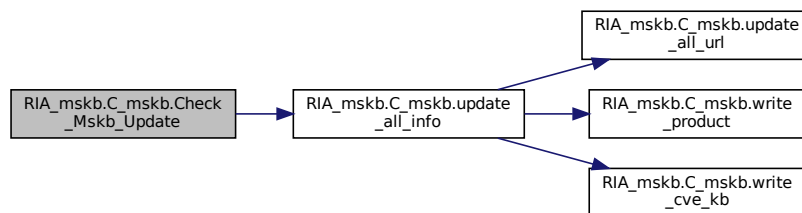
Renvoie

String d'information

Python help

Verifie s'il y des mise 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 information pour un CERTFR.

Paramètres

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

Python help

Revoie une liste pour un nom de bulletin données

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()

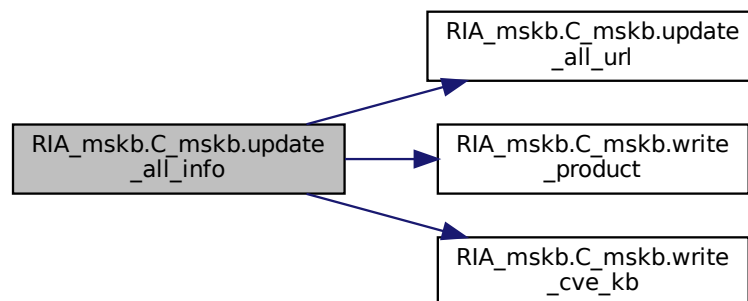
```
def RIA_mskb.C_mskb.update_all_info (
    self )
```

recupere toutes les information

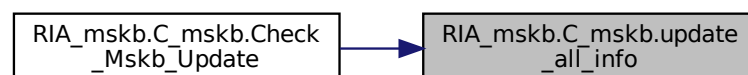
Python help

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()

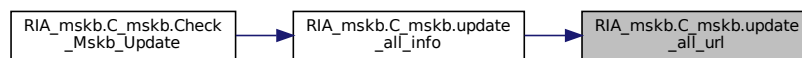
```
def RIA_mskb.C_mskb.update_all_url (
    self )
```

recupere toutes les URL

Python help

Recupère 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

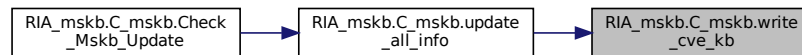
Paramètres

<i>ms_cve</i>	le CVE corrigé
<i>fix_id</i>	le numero de KB
<i>product</i>	le ProductID
<i>ms_url</i>	l'URL pour plusd'information
<i>fix_Supercedence</i>	le KB remplacé
<i>typekb</i>	le type de KB

Python help

Ecrit en BDD les information 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

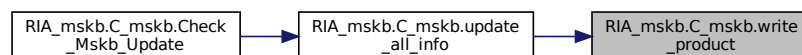
Paramètres

<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 :
[RIA_mskb.py](#)

8.5 Référence de la classe RIA_sql.C_sql

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)
ecrit dans la table information
- def `clean_tmp` (self)
efface les tables temporaires
- def `clean_new` (self)
Mets New à "0".
- def `write_sc` (self, script)
Execute un script sqlite sans retour.
- def `get_sc` (self, script)
Execute 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)
Tranferte les table tmp vers les main.
- def `flush_url` (self)
Cherche toutes CERTFR mis a jour par les URL.
- def `get_all_new_certfr` (self)
Revoie tous les bulletins mis a jour.
- def `get_certfr` (self, nom)
lit un bulletin en BDD
- def `get_all_cve_certfr` (self, certfr)
revoie 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)
revoie tous les CPE d'un bulletin
- def `get_all_cpe_uri` (self, uri)
revoie to les cpe pour un uri23
- def `get_orphan_by_obj` (self, obj)
Les bulletin par obj sans CVE.
- def `get_all_certfr_by_cve` (self)
Les bulletin par CVE.
- def `get_all_orphan` (self)
Bulletin par obj sans CVE.
- def `get_all_cve_orphan` (self)
Les CVE non present 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

8.5.1 Description détaillée

Class pour interragir avec la BDD en sqlite3

8.5.2 Documentation des constructeurs et destructeur

8.5.2.1 `__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

8.5.3 Documentation des fonctions membres

8.5.3.1 `clean_new()`

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

Mets New à "0".

Python help

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

8.5.3.2 `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

8.5.3.3 close_db()

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

fermeture

Python help

fonction de sauvegarde et fermeture

8.5.3.4 flush_tmp()

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

Tranfert les table tmp vers les main.

Python help

Transfert 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

8.5.3.5 flush_url()

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

Cherche toutes CERTFR mis a jour par les URL.

Python help

Mise a 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

8.5.3.6 `get_all_certfr_by_cve()`

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

Les bulletin par CVE.

Renvoie

liste ou []

Python help

Renvoie une liste de tous les couples CVE/Bulletins

8.5.3.7 `get_all_cpe_certfr()`

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

revoie tous les CPE d'un bulletin

Paramètres

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

Renvoie

liste de C_cpe ou None

Python help

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

8.5.3.8 `get_all_cpe_uri()`

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

revoie to les cpe pour un uri23

Paramètres

<i>uri</i>	une partie d'uri a 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
```

8.5.3.9 get_all_cve_certfr()

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

revoie tous les CVE d'un bulletin

Paramètres

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

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
```

8.5.3.10 get_all_cve_orphan()

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

Les CVE non present sur le NIST.

Renvoie

liste ou []

Python help

```
Renvoie une liste de tous les CVE des bulletins non present sur le nist
Soit ils sont pas encore valide soit le CERTFR a mal formater son bulletin
```

8.5.3.11 `get_all_new_certfr()`

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

Revoie tous les bulletins mis a jour.

Renvoie

liste

Python help

renvoie tous un liste de tous les nom de bulletin avec New=1

8.5.3.12 `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 bulletin et Objet sans CVE

8.5.3.13 `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)

8.5.3.14 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 plus clair

Python help

Revoie la date ou "" pour quoi dans la table Info

8.5.3.15 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 caractère)

A faire gérer les erreurs

Python help

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

8.5.3.16 get_orphan_by_obj()

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

Les bulletin par obj sans CVE.

Paramètres

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

Renvoie

liste ou []

Python help

Renvoie une liste des nom de bulletin et objet sans CVE
obj chaîne a chercher dans les objet des bulletins
utilise la fonction SQL LIKE

8.5.3.17 get_sc()

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

Execute un script sqlite.

Paramètres

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

Renvoie

une liste

Python help

Execute le "script" SQL et renvoie une liste de tous les resultats

8.5.3.18 load_mogs()

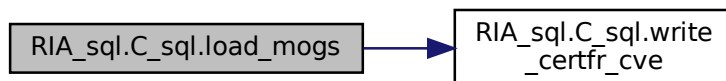
```
def RIA_sql.C_sql.load_mogs (
    self )
```

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 :



8.5.3.19 save_db()

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

Sauvegarde.

Python help

fonction de sauvegarde des transactions en cours

8.5.3.20 set_Info_date()

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

ecrit dans la table information

Paramètres

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

Python help

Sauvegarde la date pour quoi dans la table Info

8.5.3.21 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.

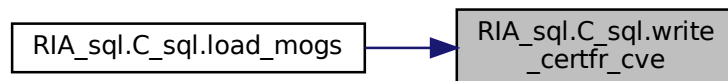
Paramètres

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

Python help

Ecrit en BDD un CVE dans la table CERTFR_tmp
calcul la Hkey UNIQUE

Voici le graphe des appelants de cette fonction :



8.5.3.22 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

8.5.3.23 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

8.5.3.24 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

8.5.3.25 write_sc()

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

Execute un script sqlite sans retour.

Paramètres

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

Python help

Execute le "script" SQL sans retour

8.5.4 Documentation des données membres

8.5.4.1 Localdb

`RIA_sql.C_sql.Localdb`

le fichier MaBdd

8.5.4.2 moncur

`RIA_sql.C_sql.moncur`

le curseur dans la Bdd

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

— [RIA_sql.py](#)

8.6 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
+ <code>__init__()</code> + <code>Reset_New()</code> + <code>Flush_cve()</code> + <code>write_url_cve()</code> + <code>Write_wrapper_info()</code> + <code>Read_wrapper_info()</code> + <code>Url_down_file()</code> + <code>Url_is_updated()</code> + <code>check_regex()</code> + <code>Download_Certfr()</code> 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 `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.
- 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

Attributs publics

- `MaBdd`
la Bdd via C_sql

8.6.1 Description détaillée

Class pour le Wrapper.

Python help

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

8.6.2 Documentation des constructeurs et destructeur

8.6.2.1 __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 C_sql déjà ouvert
On ajoute les tables SQL spécifiques
Url est UNIQUE pour gérer les conflits d'INSERT

8.6.3 Documentation des fonctions membres

8.6.3.1 Check_ALL_Wapper_Update()

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

Vérifie tous les editeurs en une seule fonction.

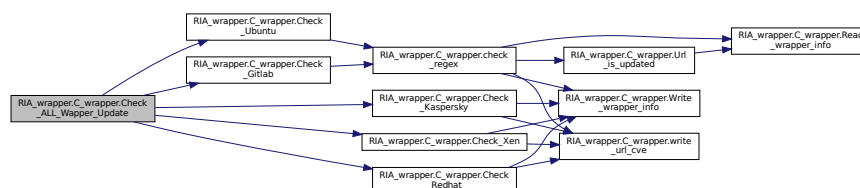
Paramètres

<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 :



8.6.3.2 Check_Gitlab()

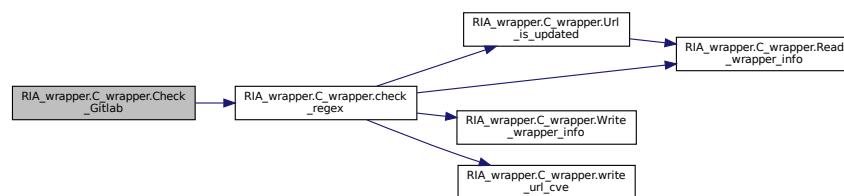
```
def RIA_wrapper.C_wrapper.Check_Gitlab (
    self )
```

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 :



8.6.3.3 Check_Kaspersky()

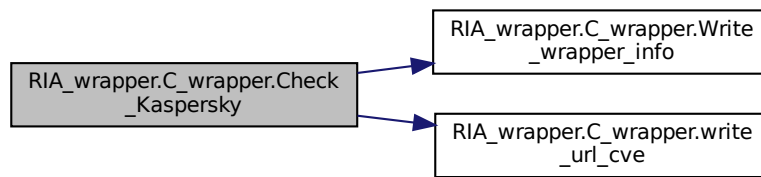
```
def RIA_wrapper.C_wrapper.Check_Kaspersky (
    self )
```

Verifie Kaspersky.

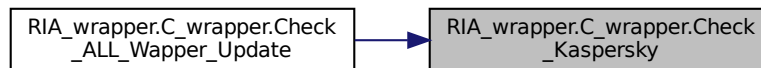
Python help

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 :



8.6.3.4 Check_Redhat()

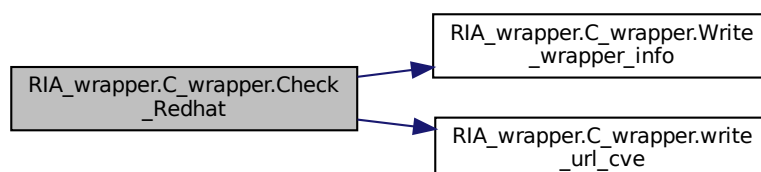
```
def RIA_wrapper.C_wrapper.Check_Redhat (
    self )
```

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 :



8.6.3.5 check_regex()

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

Parse une url en regex.

Paramètres

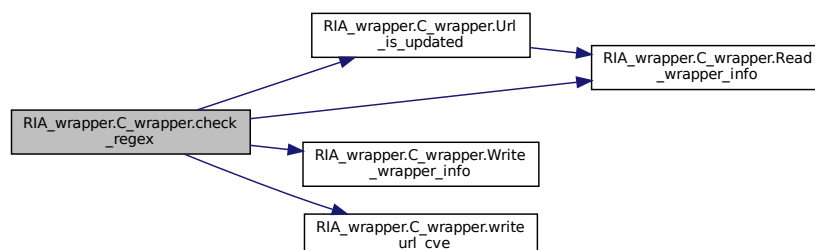
<i>info</i>	Un C_wrapper_info
-------------	-----------------------------------

Python help

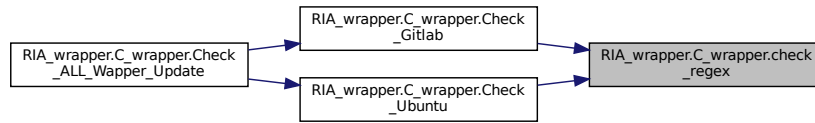
check_regex permet de parcourir une page unique avec des liens
info doit être un C_wrapper_info

- Lit la page Url
- cherche chaque sous-page avec le Regex
- parse les sous-pages pour les CVE

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.6.3.6 Check_Ubuntu()

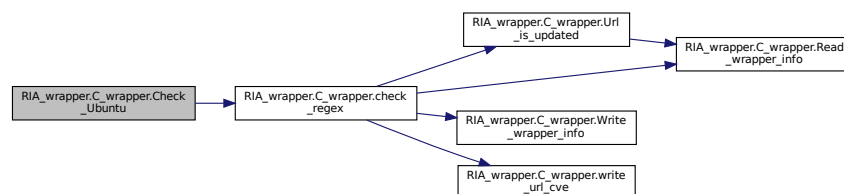
```
def RIA_wrapper.C_wrapper.Check_Ubuntu (
    self )
```

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 :



8.6.3.7 Check_Xen()

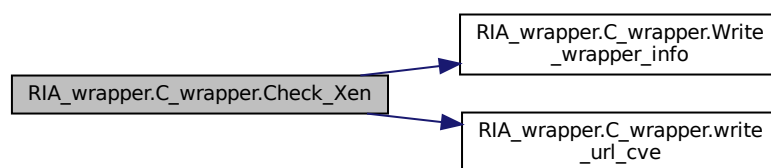
```
def RIA_wrapper.C_wrapper.Check_Xen (
    self )
```

Verifie Xen.

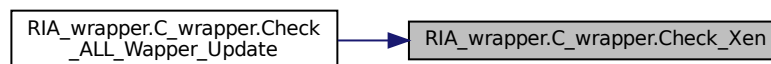
Python help

Verifie le JSON de Xen

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.6.3.8 Download_Certfr()

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

Télécharge les Tar CERTFR si plus récent.

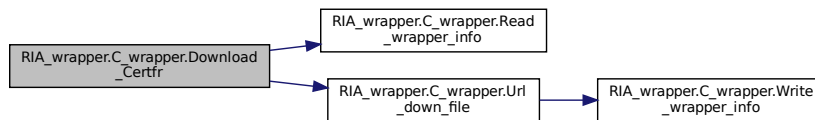
Paramètres

<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 :



8.6.3.9 Download_CVE()

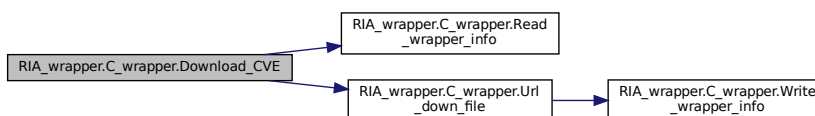
```
def RIA_wrapper.C_wrapper.Download_CVE (
    self )
```

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 :



8.6.3.10 Flush_cve()

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

Sauvegarde tous les couples Url/CVE trouvés.

Python help

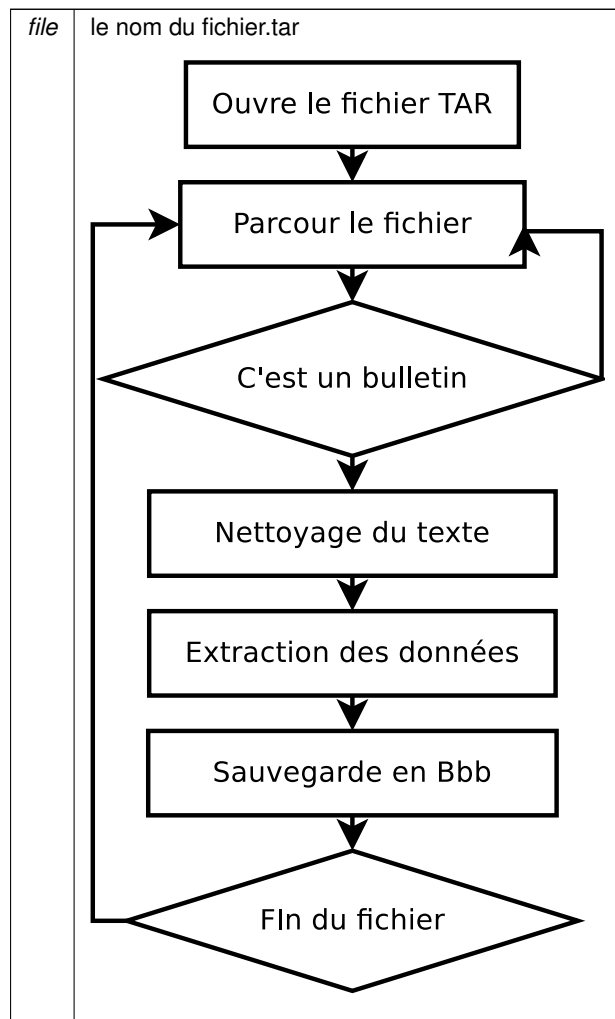
Transfert de la table TMP vers la table normale
 Selectionne tous les nom_bulletin et CVE où l'URL est commune aux deux tables
 Ajoute ensuite ces informations dans la liste officiel via write_certfr_cve

8.6.3.11 Load_TAR_certfr()

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

extrait les info CERTFR d'un TAR

Paramètres

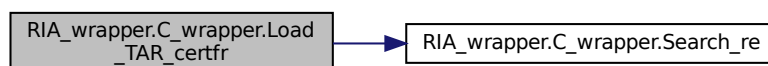


A faire nettoyer le code

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 :

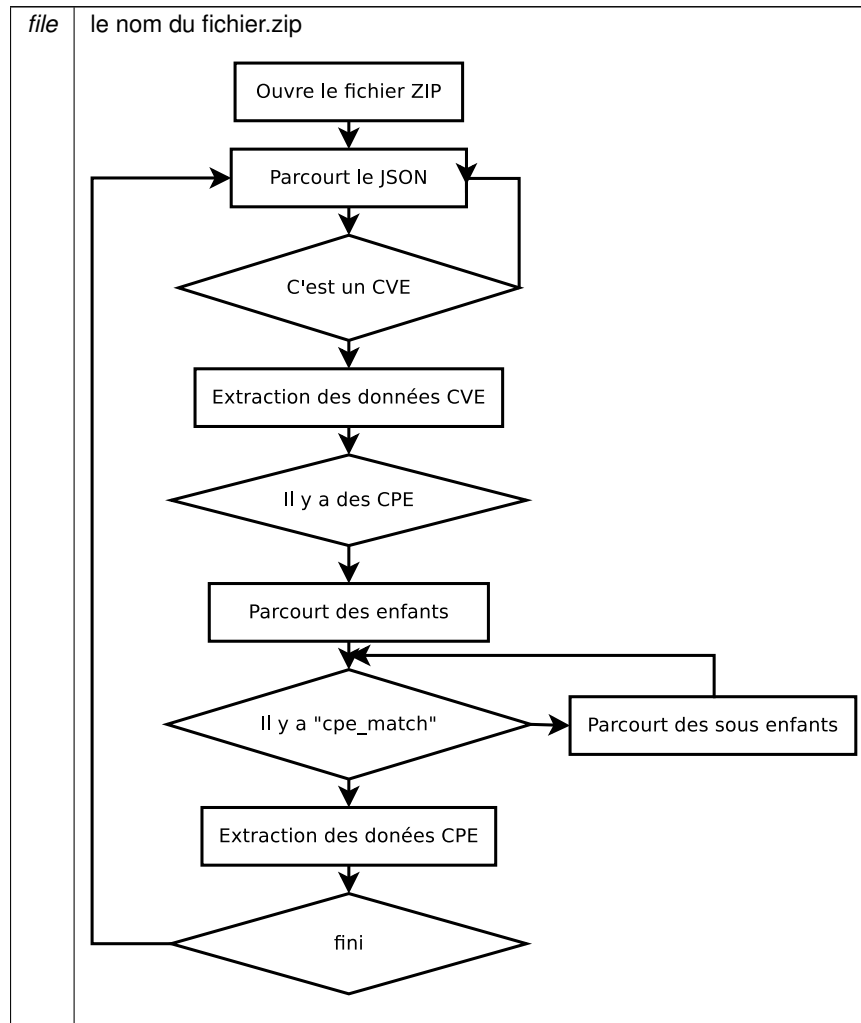


8.6.3.12 Load_ZIP_cve()

```
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"

8.6.3.13 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

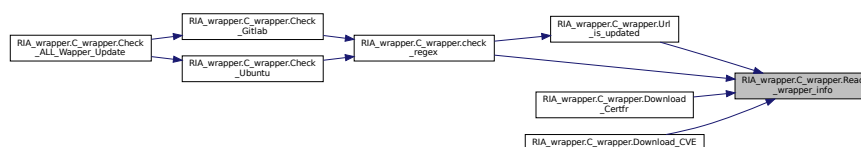
Renvoie

Une liste de [C_wrapper_info](#)

Python help

Lit une liste de 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 :



8.6.3.14 Reset_New()

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

remet a 0 le champ New

Python help

Met a 0 le champ New en BDD

8.6.3.15 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

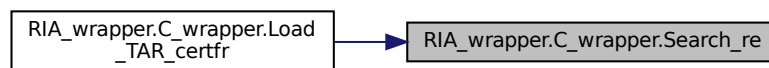
Renvoie

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 :



8.6.3.16 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

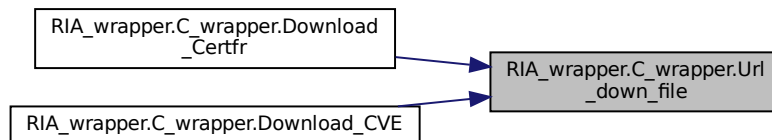
Télécharge un fichier s'il est plus récent

- 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 :



8.6.3.17 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

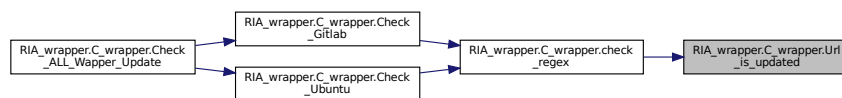
Python help

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 :



8.6.3.18 write_url_cve()

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

Sauvegarde dans URL_ck.

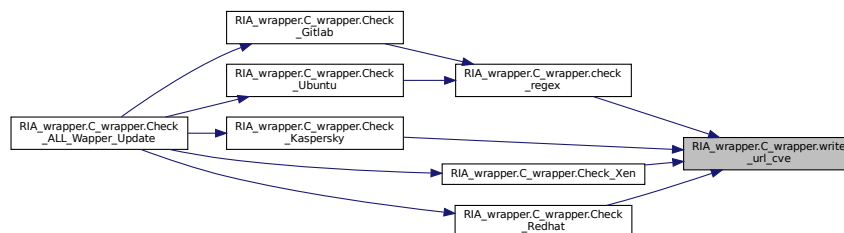
Paramètres

<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 :



8.6.3.19 Write_wrapper_info()

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

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

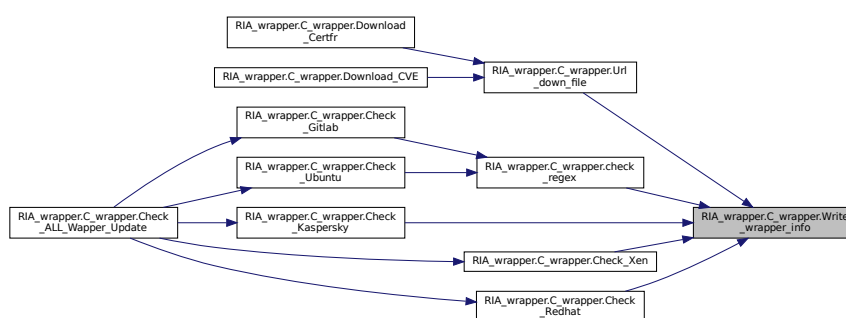
Paramètres

<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 :



8.6.4 Documentation des données membres

8.6.4.1 MaBdd

RIA_wrapper.C_wrapper.MaBdd

la Bdd via C_sql

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

— [RIA_wrapper.py](#)

8.7 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.

8.7.1 Description détaillée

Class Objet info Wrapper.

Python help

Objet pour manipuler les information sans passer par des listes

8.7.2 Documentation des constructeurs et destructeur

8.7.2.1 `__init__()`

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

le constructor

Python help

constructor

8.7.3 Documentation des données membres

8.7.3.1 Date

```
RIA_wrapper.C_wrapper_info.Date
```

Date de référence.

8.7.3.2 Fichier

```
RIA_wrapper.C_wrapper_info.Fichier
```

Le fichier sert pour les CVE et CERTFR.

8.7.3.3 Module

```
RIA_wrapper.C_wrapper_info.Module
```

Quelle fonction fait la recherche.

8.7.3.4 New

```
RIA_wrapper.C_wrapper_info.New
```

Boolean Sqlite si nouveau.

8.7.3.5 Regex

`RIA_wrapper.C_wrapper_info.Regex`

la Regex de recherche de lien dans la page

8.7.3.6 Rep

`RIA_wrapper.C_wrapper_info.Rep`

Le sous répertoire de sauvegarde du fichier.

8.7.3.7 S_Url

`RIA_wrapper.C_wrapper_info.S_Url`

le prefix a ajouter aux liens trouvés

8.7.3.8 Taille

`RIA_wrapper.C_wrapper_info.Taille`

La taille du fichier sans emploi maintenant.

8.7.3.9 Url

`RIA_wrapper.C_wrapper_info.Url`

L'url.

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

— [RIA_wrapper.py](#)

Chapitre 9

Documentation des fichiers

9.1 Référence du fichier README.md

9.2 Référence du fichier RIA.py

Recherche d'Information Automatisée.

Espaces de nommage

— [RIA](#)

Fonctions

```
— def RIA.credit ()  
    Affiche simplement le credit.  
— def RIA.CERT\_to\_STR (Nom, tab)  
    Une jolie sortie formater des info CERTFR.  
— def RIA.MS\_to\_STR (Nom, tab)  
    Une jolie sortie formater des info Microsoft.  
— def RIA.Write\_CERTFR (nom, annee)  
    Ecrit dans un fichier text les informations du bulletin.  
— def RIA.URI\_to\_FILE (Nom, uri)  
    Ecrit un fichiers avec bulletins et URI23 pour une recherche.  
— def RIA.mon\_script ()  
    Core du scripts.
```

9.2.1 Description détaillée

Recherche d'Information Automatisée.

Auteur

Frack113

Date

07/04/2020

A faire utiliser les best practice Python

9.3 Référence du fichier 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](#)

9.3.1 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 des bulletins,CVE ou CPE sans passer par le sql

9.4 Référence du fichier RIA_mskb.py

La gestion de l'API Microsoft.

Classes

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

Espaces de nommage

- [RIA_mskb](#)

9.4.1 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 cab <http://go.microsoft.com/fwlink/?linkid=74689>

9.5 Référence du fichier RIA_sql.py

La gestion de la base de données.

Classes

— class [RIA_sql.C_sql](#)

Espaces de nommage

— [RIA_sql](#)

9.5.1 Description détaillée

La gestion de la base de données.

Auteur

Frack113

Date

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

Python help

9.6 Référence du fichier RIA_wrapper.py

La gestion des wrapper 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](#)

9.6.1 Description détaillée

La gestion des wrapper Internet.

Auteur

Frack113

Date

08/04/2020 module pour les recherches Internet

Index

- __init__
 - RIA_class.C_certfr, [20](#)
 - RIA_class.C_cpe, [25](#)
 - RIA_class.C_cve, [29](#)
 - RIA_mskb.C_mskb, [33](#)
 - RIA_sql.C_sql, [40](#)
 - RIA_wrapper.C_wrapper, [51](#)
 - RIA_wrapper.C_wrapper_info, [67](#)
- api_key
 - RIA_mskb.C_mskb, [37](#)
- api_type
 - RIA_mskb.C_mskb, [37](#)
- api_url
 - RIA_mskb.C_mskb, [38](#)
- CERT_to_STR
 - RIA, [13](#)
- Check_ALL_Wapper_Update
 - RIA_wrapper.C_wrapper, [52](#)
- Check_Gitlab
 - RIA_wrapper.C_wrapper, [52](#)
- Check_Kaspersky
 - RIA_wrapper.C_wrapper, [53](#)
- Check_Mskb_Update
 - RIA_mskb.C_mskb, [33](#)
- Check_Redhat
 - RIA_wrapper.C_wrapper, [54](#)
- check_regex
 - RIA_wrapper.C_wrapper, [55](#)
- Check_Ubuntu
 - RIA_wrapper.C_wrapper, [56](#)
- Check_Xen
 - RIA_wrapper.C_wrapper, [56](#)
- clean_new
 - RIA_sql.C_sql, [40](#)
- clean_tmp
 - RIA_sql.C_sql, [40](#)
- close_db
 - RIA_sql.C_sql, [40](#)
- conf
 - RIA_class.C_cpe, [26](#)
- cpe23Uri
 - RIA_class.C_cpe, [26](#)
- crc
 - RIA_class.C_certfr, [22](#)
 - RIA_class.C_cpe, [26](#)
 - RIA_class.C_cve, [30](#)
- credit
 - RIA, [14](#)
- cve
 - RIA_class.C_cpe, [26](#)
- cvssV2
 - RIA_class.C_cve, [30](#)
- cvssV2base
 - RIA_class.C_cve, [30](#)
- cvssV3
 - RIA_class.C_cve, [31](#)
- cvssV3base
 - RIA_class.C_cve, [31](#)
- Date
 - RIA_wrapper.C_wrapper_info, [67](#)
- dateOrigine
 - RIA_class.C_certfr, [22](#)
 - RIA_class.C_cve, [31](#)
- dateUpdate
 - RIA_class.C_certfr, [23](#)
 - RIA_class.C_cve, [31](#)
- decode_file
 - RIA_class.C_certfr, [21](#)
- decode_link
 - RIA_class.C_certfr, [21](#)
- Download_Certfr
 - RIA_wrapper.C_wrapper, [57](#)
- Download_CVE
 - RIA_wrapper.C_wrapper, [58](#)
- encode_file
 - RIA_class.C_certfr, [21](#)
- encode_link
 - RIA_class.C_certfr, [21](#)
- Fichier
 - RIA_wrapper.C_wrapper_info, [67](#)
- file
 - RIA_class.C_certfr, [23](#)
- Flush_cve
 - RIA_wrapper.C_wrapper, [58](#)
- flush_tmp
 - RIA_sql.C_sql, [41](#)
- flush_url
 - RIA_sql.C_sql, [41](#)
- get_all_certfr_by_cve
 - RIA_sql.C_sql, [41](#)
- get_all_cpe_certfr
 - RIA_sql.C_sql, [42](#)
- get_all_cpe_uri
 - RIA_sql.C_sql, [42](#)

- get_all_cve_certfr
 - RIA_sql.C_sql, [43](#)
- get_all_cve_orphan
 - RIA_sql.C_sql, [43](#)
- get_all_new_certfr
 - RIA_sql.C_sql, [43](#)
- get_all_orphan
 - RIA_sql.C_sql, [44](#)
- get_certfr
 - RIA_sql.C_sql, [44](#)
- get_info_certfr
 - RIA_mskb.C_mskb, [34](#)
- get_Info_date
 - RIA_sql.C_sql, [45](#)
- get_max_lg_uri_cpe
 - RIA_sql.C_sql, [45](#)
- get_orphan_by_obj
 - RIA_sql.C_sql, [45](#)
- get_sc
 - RIA_sql.C_sql, [46](#)
- header
 - RIA_mskb.C_mskb, [38](#)
- id
 - RIA_class.C_cpe, [27](#)
 - RIA_class.C_cve, [31](#)
- link
 - RIA_class.C_certfr, [23](#)
- load_mogs
 - RIA_sql.C_sql, [46](#)
- Load_TAR_certfr
 - RIA_wrapper.C_wrapper, [58](#)
- Load_ZIP_cve
 - RIA_wrapper.C_wrapper, [59](#)
- Localdb
 - RIA_sql.C_sql, [50](#)
- MaBdd
 - RIA_mskb.C_mskb, [38](#)
 - RIA_wrapper.C_wrapper, [65](#)
- Module
 - RIA_wrapper.C_wrapper_info, [67](#)
- mon_script
 - RIA, [14](#)
- moncur
 - RIA_sql.C_sql, [50](#)
- MS_to_STR
 - RIA, [14](#)
- New
 - RIA_class.C_certfr, [23](#)
 - RIA_class.C_cpe, [27](#)
 - RIA_class.C_cve, [31](#)
 - RIA_wrapper.C_wrapper_info, [67](#)
- nom
 - RIA_class.C_certfr, [23](#)
- obj
 - RIA_class.C_certfr, [23](#)
- opérateur
 - RIA_class.C_cpe, [27](#)
- Read_wrapper_info
 - RIA_wrapper.C_wrapper, [60](#)
- README.md, [69](#)
- Regex
 - RIA_wrapper.C_wrapper_info, [67](#)
- Rep
 - RIA_wrapper.C_wrapper_info, [68](#)
- reset
 - RIA_class.C_certfr, [22](#)
 - RIA_class.C_cpe, [25](#)
 - RIA_class.C_cve, [30](#)
- reset_db
 - RIA_mskb.C_mskb, [35](#)
- Reset_New
 - RIA_wrapper.C_wrapper, [61](#)
- RIA, [13](#)
 - CERT_to_STR, [13](#)
 - credit, [14](#)
 - mon_script, [14](#)
 - MS_to_STR, [14](#)
 - URI_to_FILE, [15](#)
 - Write_CERTFR, [16](#)
- RIA.py, [69](#)
- RIA_class, [17](#)
- RIA_class.C_certfr, [19](#)
 - __init__, [20](#)
 - crc, [22](#)
 - dateOrigine, [22](#)
 - dateUpdate, [23](#)
 - decode_file, [21](#)
 - decode_link, [21](#)
 - encode_file, [21](#)
 - encode_link, [21](#)
 - file, [23](#)
 - link, [23](#)
 - New, [23](#)
 - nom, [23](#)
 - obj, [23](#)
 - reset, [22](#)
 - set_crc, [22](#)
- RIA_class.C_cpe, [24](#)
 - __init__, [25](#)
 - conf, [26](#)
 - cpe23Uri, [26](#)
 - crc, [26](#)
 - cve, [26](#)
 - id, [27](#)
 - New, [27](#)
 - opérateur, [27](#)
 - reset, [25](#)
 - set_crc, [26](#)
 - versionEndExcluding, [27](#)
 - versionEndIncluding, [27](#)
 - versionStartExcluding, [27](#)
 - versionStartIncluding, [28](#)

- vulnerable, [28](#)
- RIA_class.C_cve, [28](#)
 - __init__, [29](#)
 - crc, [30](#)
 - cvssV2, [30](#)
 - cvssV2base, [30](#)
 - cvssV3, [31](#)
 - cvssV3base, [31](#)
 - dateOrigine, [31](#)
 - dateUpdate, [31](#)
 - id, [31](#)
 - New, [31](#)
 - reset, [30](#)
 - set_crc, [30](#)
- RIA_class.py, [70](#)
- RIA_mskb, [17](#)
- RIA_mskb.C_mskb, [32](#)
 - __init__, [33](#)
 - api_key, [37](#)
 - api_type, [37](#)
 - api_url, [38](#)
 - Check_Mskb_Update, [33](#)
 - get_info_certfr, [34](#)
 - header, [38](#)
 - MaBdd, [38](#)
 - reset_db, [35](#)
 - update_all_info, [35](#)
 - update_all_url, [35](#)
 - write_cve_kb, [36](#)
 - write_product, [37](#)
- RIA_mskb.py, [70](#)
- RIA_sql, [17](#)
- RIA_sql.C_sql, [38](#)
 - __init__, [40](#)
 - clean_new, [40](#)
 - clean_tmp, [40](#)
 - close_db, [40](#)
 - flush_tmp, [41](#)
 - flush_url, [41](#)
 - get_all_certfr_by_cve, [41](#)
 - get_all_cpe_certfr, [42](#)
 - get_all_cpe_uri, [42](#)
 - get_all_cve_certfr, [43](#)
 - get_all_cve_orphan, [43](#)
 - get_all_new_certfr, [43](#)
 - get_all_orphan, [44](#)
 - get_certfr, [44](#)
 - get_Info_date, [45](#)
 - get_max_lg_uri_cpe, [45](#)
 - get_orphan_by_obj, [45](#)
 - get_sc, [46](#)
 - load_mogs, [46](#)
 - Localdb, [50](#)
 - moncur, [50](#)
 - save_db, [47](#)
 - set_Info_date, [47](#)
 - write_certfr_cve, [47](#)
 - write_certfr_tmp, [48](#)
 - write_cpe_tmp, [48](#)
 - write_cve_tmp, [49](#)
 - write_sc, [49](#)
- RIA_sql.py, [71](#)
- RIA_wrapper, [17](#)
- RIA_wrapper.C_wrapper, [50](#)
 - __init__, [51](#)
 - Check_ALL_Wapper_Update, [52](#)
 - Check_Gitlab, [52](#)
 - Check_Kaspersky, [53](#)
 - Check_Redhat, [54](#)
 - check_regex, [55](#)
 - Check_Ubuntu, [56](#)
 - Check_Xen, [56](#)
 - Download_Certfr, [57](#)
 - Download_CVE, [58](#)
 - Flush_cve, [58](#)
 - Load_TAR_certfr, [58](#)
 - Load_ZIP_cve, [59](#)
 - MaBdd, [65](#)
 - Read_wrapper_info, [60](#)
 - Reset_New, [61](#)
 - Search_re, [61](#)
 - Url_down_file, [62](#)
 - Url_is_updated, [63](#)
 - write_url_cve, [64](#)
 - Write_wrapper_info, [64](#)
- RIA_wrapper.C_wrapper_info, [66](#)
 - __init__, [67](#)
 - Date, [67](#)
 - Fichier, [67](#)
 - Module, [67](#)
 - New, [67](#)
 - Regex, [67](#)
 - Rep, [68](#)
 - S_Url, [68](#)
 - Taille, [68](#)
 - Url, [68](#)
- RIA_wrapper.py, [71](#)
- S_Url
 - RIA_wrapper.C_wrapper_info, [68](#)
- save_db
 - RIA_sql.C_sql, [47](#)
- Search_re
 - RIA_wrapper.C_wrapper, [61](#)
- set_crc
 - RIA_class.C_certfr, [22](#)
 - RIA_class.C_cpe, [26](#)
 - RIA_class.C_cve, [30](#)
- set_Info_date
 - RIA_sql.C_sql, [47](#)
- Taille
 - RIA_wrapper.C_wrapper_info, [68](#)
- update_all_info
 - RIA_mskb.C_mskb, [35](#)
- update_all_url

- RIA_mskb.C_mskb, [35](#)
- URI_to_FILE
 - RIA, [15](#)
- Url
 - RIA_wrapper.C_wrapper_info, [68](#)
- Url_down_file
 - RIA_wrapper.C_wrapper, [62](#)
- Url_is_updated
 - RIA_wrapper.C_wrapper, [63](#)
- versionEndExcluding
 - RIA_class.C_cpe, [27](#)
- versionEndIncluding
 - RIA_class.C_cpe, [27](#)
- versionStartExcluding
 - RIA_class.C_cpe, [27](#)
- versionStartIncluding
 - RIA_class.C_cpe, [28](#)
- vulnerable
 - RIA_class.C_cpe, [28](#)
- Write_CERTFR
 - RIA, [16](#)
- write_certfr_cve
 - RIA_sql.C_sql, [47](#)
- write_certfr_tmp
 - RIA_sql.C_sql, [48](#)
- write_cpe_tmp
 - RIA_sql.C_sql, [48](#)
- write_cve_kb
 - RIA_mskb.C_mskb, [36](#)
- write_cve_tmp
 - RIA_sql.C_sql, [49](#)
- write_product
 - RIA_mskb.C_mskb, [37](#)
- write_sc
 - RIA_sql.C_sql, [49](#)
- write_url_cve
 - RIA_wrapper.C_wrapper, [64](#)
- Write_wrapper_info
 - RIA_wrapper.C_wrapper, [64](#)