

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	20
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()	21
8.1.3.5 reset()	21
8.1.3.6 set_crc()	21
8.1.4 Documentation des données membres	22
8.1.4.1 crc	22

8.1.4.2 dateOrigine	22
8.1.4.3 dateUpdate	22
8.1.4.4 file	22
8.1.4.5 link	22
8.1.4.6 New	23
8.1.4.7 nom	23
8.1.4.8 obj	23
8.2 Référence de la classe RIA_class.C_cpe	23
8.2.1 Description détaillée	24
8.2.2 Documentation des constructeurs et destructeur	24
8.2.2.1 __init__()	24
8.2.3 Documentation des fonctions membres	24
8.2.3.1 reset()	25
8.2.3.2 set_crc()	25
8.2.4 Documentation des données membres	25
8.2.4.1 conf	25
8.2.4.2 cpe23Uri	25
8.2.4.3 crc	25
8.2.4.4 cve	26
8.2.4.5 id	26
8.2.4.6 New	26
8.2.4.7 operateur	26
8.2.4.8 versionEndExcluding	26
8.2.4.9 versionEndIncluding	26
8.2.4.10 versionStartExcluding	27
8.2.4.11 versionStartIncluding	27
8.2.4.12 vulnerable	27
8.3 Référence de la classe RIA_class.C_cve	27
8.3.1 Description détaillée	28
8.3.2 Documentation des constructeurs et destructeur	28
8.3.2.1 __init__()	28
8.3.3 Documentation des fonctions membres	28
8.3.3.1 reset()	29
8.3.3.2 set_crc()	29
8.3.4 Documentation des données membres	29
8.3.4.1 crc	29
8.3.4.2 cvssV2	29
8.3.4.3 cvssV2base	29
8.3.4.4 cvssV3	30
8.3.4.5 cvssV3base	30
8.3.4.6 dateOrigine	30
8.3.4.7 dateUpdate	30

8.3.4.8 id	30
8.3.4.9 New	30
8.4 Référence de la classe RIA_mskb.C_mskb	31
8.4.1 Description détaillée	32
8.4.2 Documentation des constructeurs et destructeur	32
8.4.2.1 __init__()	32
8.4.3 Documentation des fonctions membres	32
8.4.3.1 Check_Mskb_Update()	32
8.4.3.2 get_info_certfr()	33
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()	35
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	37
8.5 Référence de la classe RIA_sql.C_sql	37
8.5.1 Documentation des constructeurs et destructeur	38
8.5.1.1 __init__()	38
8.5.2 Documentation des fonctions membres	38
8.5.2.1 clean_new()	39
8.5.2.2 clean_tmp()	39
8.5.2.3 close_db()	39
8.5.2.4 flush_tmp()	39
8.5.2.5 get_all_certfr_by_cve()	39
8.5.2.6 get_all_cpe_certfr()	39
8.5.2.7 get_all_cpe_uri()	40
8.5.2.8 get_all_cve_certfr()	40
8.5.2.9 get_all_new_certfr()	41
8.5.2.10 get_all_new_url()	41
8.5.2.11 get_all_orphan()	41
8.5.2.12 get_certfr()	41
8.5.2.13 get_Info_date()	42
8.5.2.14 get_max_lg_uri_cpe()	42
8.5.2.15 get_orphan_by_obj()	42
8.5.2.16 get_sc()	43
8.5.2.17 get_url_info()	43
8.5.2.18 load_mogs()	43

8.5.2.19 save_db()	44
8.5.2.20 set_Info_date()	44
8.5.2.21 set_url_info()	44
8.5.2.22 write_certfr_cve()	45
8.5.2.23 write_certfr_tmp()	45
8.5.2.24 write_cpe_tmp()	46
8.5.2.25 write_cve_tmp()	46
8.5.2.26 write_sc()	46
8.5.3 Documentation des données membres	46
8.5.3.1 Localdb	47
8.5.3.2 moncur	47
8.6 Référence de la classe RIA_wrapper.C_wrapper	47
8.6.1 Description détaillée	48
8.6.2 Documentation des constructeurs et destructeur	48
8.6.2.1 __init__()	48
8.6.3 Documentation des fonctions membres	49
8.6.3.1 Check_ALL_Wapper_Update()	49
8.6.3.2 Check_Gitlab()	49
8.6.3.3 Check_Kaspersky()	50
8.6.3.4 check_regex()	50
8.6.3.5 Check_Ubuntu()	51
8.6.3.6 Check_Xen()	52
8.6.3.7 Download_Certfr()	52
8.6.3.8 Download_CVE()	53
8.6.3.9 Flush_cve()	53
8.6.3.10 Load_TAR_certfr()	53
8.6.3.11 Load_ZIP_cve()	54
8.6.3.12 Read_wrapper_info()	55
8.6.3.13 Reset_New()	56
8.6.3.14 Search_re()	56
8.6.3.15 Url_down_file()	57
8.6.3.16 Url_is_updated()	57
8.6.3.17 write_url_cve()	58
8.6.3.18 Write_wrapper_info()	59
8.6.4 Documentation des données membres	59
8.6.4.1 MaBdd	59
8.7 Référence de la classe RIA_wrapper.C_wrapper_info	60
8.7.1 Description détaillée	60
8.7.2 Documentation des constructeurs et destructeur	60
8.7.2.1 __init__()	61
8.7.3 Documentation des données membres	61
8.7.3.1 Date	61

8.7.3.2 Fichier	61
8.7.3.3 Module	61
8.7.3.4 New	61
8.7.3.5 Regex	61
8.7.3.6 Rep	61
8.7.3.7 S_Url	62
8.7.3.8 Taille	62
8.7.3.9 Url	62
9 Documentation des fichiers	63
9.1 Référence du fichier README.md	63
9.2 Référence du fichier RIA.py	63
9.2.1 Description détaillée	63
9.3 Référence du fichier RIA_class.py	64
9.3.1 Description détaillée	64
9.4 Référence du fichier RIA_mskb.py	64
9.4.1 Description détaillée	65
9.5 Référence du fichier RIA_sql.py	65
9.5.1 Description détaillée	65
9.6 Référence du fichier RIA_wrapper.py	65
9.6.1 Description détaillée	66
Index	67

Chapitre 1

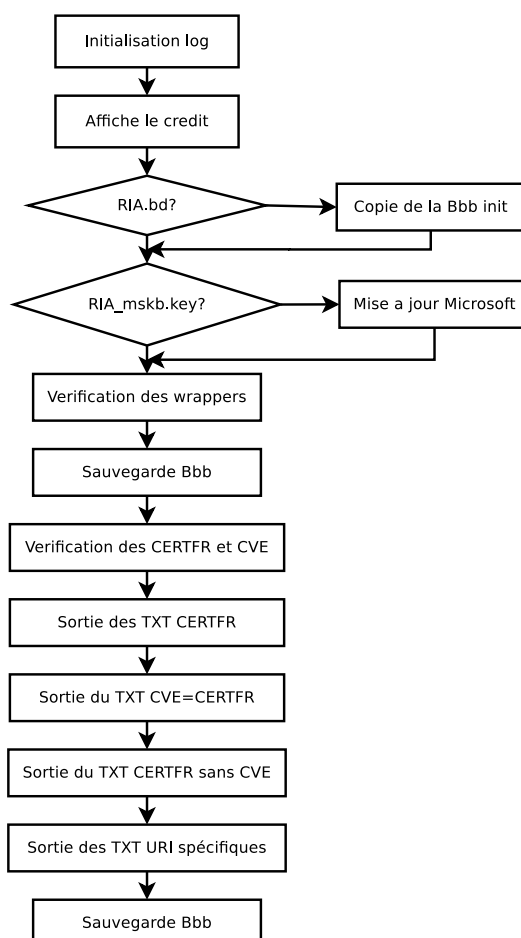
Page principale

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

- Telechargement automatique des CERTFR et CVE
- Lecture des bulletin CERTFR depuis archive rar
- Lecture des cve depuis les json
- Gestion en SQLITE
- Liste des KB via **API** microsoft
- Wrapper site éditeur pour trouver les CVE
- Ajout de l'aide des **Mogs** (ajout manuel)

Todo

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

Doculmentation

- Doxygen
- Graphviz
- Textlive pour LATEX to PDF

Chapitre 3

Liste des choses à faire

Membre `RIA.mon_script()`

simplifier les repetition wrapper

Fichier `RIA.py`

utiliser les best practice Python

Fichier `RIA_mskb.py`

optimiser le code

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

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

voir pour revoie [] au lieu de None

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

voir pour revoie [] au lieu de None

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

voir pour revoie [] au lieu de None

Membre `RIA_wrapper.C_wrapper.Flush_cve(self)`

traite les New

Membre `RIA_wrapper.C_wrapper.Load_TAR_certfr(file)`

nettoyer le code

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

nettoyer le code

Fichier `RIA_wrapper.py`

traiter les New

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	23
RIA_class.C_cve	
Manipulation des CVE	27
RIA_mskb.C_mskb	
Gestion API Microsoft	31
RIA_sql.C_sql	37
RIA_wrapper.C_wrapper	
Class pour le Wrapper	47
RIA_wrapper.C_wrapper_info	
Class Objet info Wrapper	60

Chapitre 6

Index des fichiers

6.1 Liste des fichiers

Liste de tous les fichiers avec une brève description :

RIA.py	Mon IA	63
RIA_class.py	Gestion des CERTFR,CVE et CPE	64
RIA_mskb.py	La gestion de l'API Microsoft	64
RIA_sql.py	La gestion de la base de données	65
RIA_wrapper.py	La gestion des wrapper Internet	65

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.

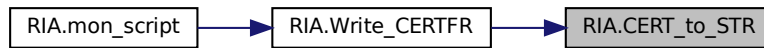
Paramètres

<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()**

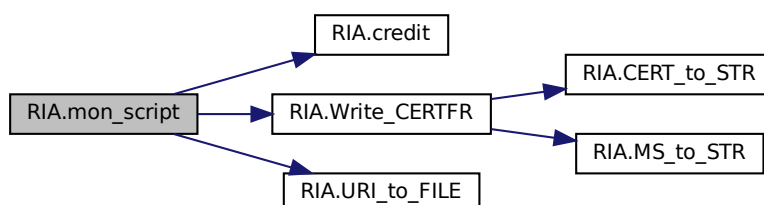
```
def RIA.mon_script ( )
```

Core du scripts.

le coeur du scripts

A faire simplifier les repetition wrapper

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.

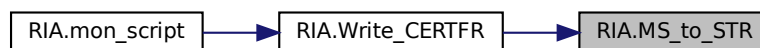
Paramètres

<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()

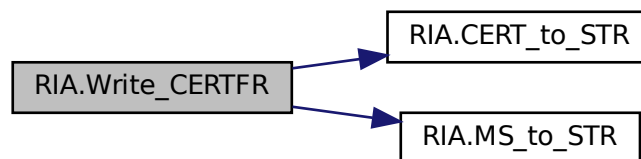
```
def RIA.Write_CERTFR (
    nom,
    annee )
```

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 surement plus utile
- `def decode_link (self)`
decode les liens en base64 surement plus utile

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.

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.

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

8.1.3.2 decode_link()

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

decode les liens en base64 surement plus utile

8.1.3.3 encode_file()

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

encode le fichier en base64

8.1.3.4 encode_link()

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

encode les liens en base64 surement plus utile

8.1.3.5 reset()

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

permet de remettre les variables à l'état initial

8.1.3.6 set_crc()

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

Calcul le CRC pour la clé UNIQUE SQL.

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.

Graphes de collaboration de RIA_class.C_cpe:

RIA_class.C_cpe
+ id + cve + conf + operateur + vulnerable + cpe23Uri + versionStartExcluding + versionStartIncluding + versionEndExcluding + versionEndIncluding + New + crc
+ __init__() + reset() + set_crc()

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.

8.2.2 Documentation des constructeurs et destructeur

8.2.2.1 __init__()

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

The 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.

8.2.3.2 set_crc()

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

Calcul le CRC pour la clé UNIQUE SQL.

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 incluse.

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)
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 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

8.3.2 Documentation des constructeurs et destructeur

8.3.2.1 `__init__()`

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

The 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.

8.3.3.2 set_crc()

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

Calcul le CRC pour la clé UNIQUE SQL.

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.

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
--------------	----------------

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.

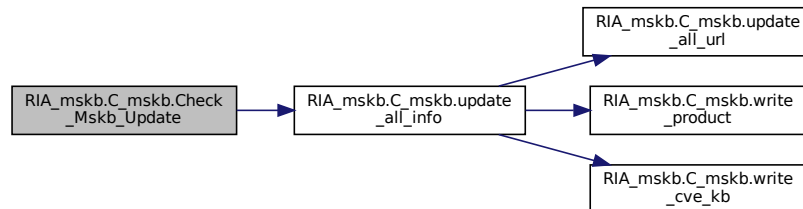
Paramètres

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

Renvoie

String d'information

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
---------------	------------------

8.4.3.3 reset_db()

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

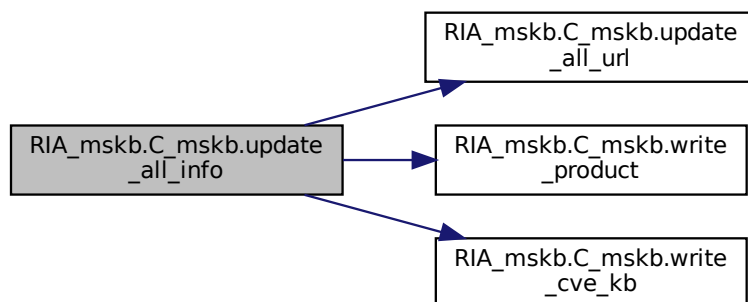
efface totalement les tables

8.4.3.4 update_all_info()

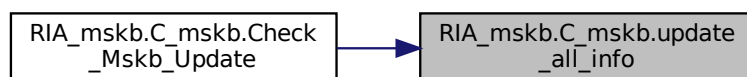
```
def RIA_mskb.C_mskb.update_all_info (
    self )
```

recupere toutes les information

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :

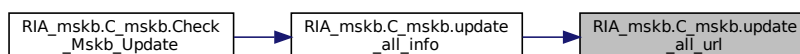


8.4.3.5 update_all_url()

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

recupere toutes les URL

Voici le graphe des appelants de cette fonction :



8.4.3.6 write_cve_kb()

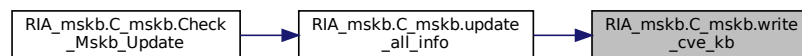
```
def RIA_mskb.C_mskb.write_cve_kb (
    self,
    ms_cve,
    fix_id,
    product,
    ms_url,
    fix_Supercedence,
    typekb )
```

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

Voici le graphe des appelants de cette fonction :



8.4.3.7 write_product()

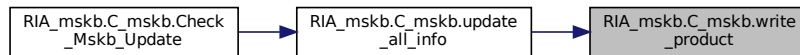
```
def RIA_mskb.C_mskb.write_product (
    self,
    ProductID,
    Value )
```

sauvegarde en BDD un ProductID

Paramètres

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

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 17 de plus...

Fonctions membres publiques

- def [__init__](#) (self)
constructors
- def [close_db](#) (self)
On ferme proprement.
- 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)
Met a '0' tous les champs New.
- 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 dans la table CERTFR_tmp.
- def `write_certfr_cve` (self, certfr, cve)
Ecrit dans la table CERTFR_tmp calcul le CRC automatiquement.
- def `write_cve_tmp` (self, moncve)
Ecrit dans la table CVE_tmp.
- def `write_cpe_tmp` (self, moncpe)
Ecrit dans la table CVE_cpe_tmp.
- def `flush_tmp` (self)
Tranfert les table tmp vers les main gestion des doublons : 1 Hkey déjà présente 2 suppression des anciens enregistrement.
- def `get_url_info` (self, nom)
Cherche l'url d'un fichiers.
- def `get_all_new_url` (self)
Cherche toutes les url modifier.
- def `set_url_info` (self, nom, date, taille, scr)
Sauvegarde.
- def `get_all_new_certfr` (self)
Revoie tous les bulletins mis a jour.
- def `get_certfr` (self, nom)
lit un bulletin
- def `get_all_cve_certfr` (self, certfr)
revoie tous les CVE d'un bulletin
- def `get_max_lg_uri_cpe` (self, certfr)
donne la taille max des uri23
- 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 recherche sql like uri%
- def `get_orphan_by_obj` (self, obj)
revoie les bulletin par obj sans CVE
- def `get_all_certfr_by_cve` (self)
revoie les bulletin par CVE
- def `get_all_orphan` (self)
revoie les bulletin par obj sans CVE
- 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 Documentation des constructeurs et destructeur

8.5.1.1 `__init__()`

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

constructors

8.5.2 Documentation des fonctions membres

8.5.2.1 clean_new()

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

Met a '0' tous les champs New.

8.5.2.2 clean_tmp()

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

efface les tables temporaires

8.5.2.3 close_db()

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

On ferme proprement.

8.5.2.4 flush_tmp()

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

Tranfert les table tmp vers les main gestion des doublons : 1 Hkey déjà présente 2 suppression des anciens enregistrement.

8.5.2.5 get_all_certfr_by_cve()

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

revoie les bulletin par CVE

Renvoie

liste ou []

8.5.2.6 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

A faire voir pour revoie [] au lieu de None

8.5.2.7 get_all_cpe_uri()

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

revoie to les cpe pour un uri23 recherche sql like uri%

Paramètres

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

Renvoie

liste de C_cpe ou None

A faire voir pour revoie [] au lieu de None

8.5.2.8 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 None

A faire voir pour revoie [] au lieu de None

8.5.2.9 get_all_new_certfr()

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

Revoie tous les bulletins mis a jour.

Renvoie

liste

8.5.2.10 get_all_new_url()

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

Cherche toutes les url modifier.

Renvoie

une liste si pas de mise a jour []

8.5.2.11 get_all_orphan()

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

revoie les bulletin par obj sans CVE

Renvoie

liste ou []

8.5.2.12 get_certfr()

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

lit un bulletin

Paramètres

nom	
-----	--

Renvoie

C_certfr

8.5.2.13 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
-------------	-----------------------

8.5.2.14 get_max_lg_uri_cpe()

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

donne la taille max des uri23

Paramètres

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

Renvoie

la taille (nb de caractère)

8.5.2.15 get_orphan_by_obj()

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

revoie les bulletin par obj sans CVE

Paramètres

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

Renvoie

liste ou []

8.5.2.16 get_sc()

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

Execute un script sqlite.

Paramètres

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

Renvoie

une liste

8.5.2.17 get_url_info()

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

Cherche l'url d'un fichiers.

Paramètres

<i>nom</i>	Le nom du fichier
------------	-------------------

Renvoie

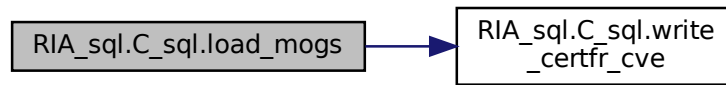
une liste si pas present []

8.5.2.18 load_mogs()

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

Charge en BDD les couples CERTFR;CVE trouvés manuellement.

Voici le graphe d'appel pour cette fonction :



8.5.2.19 save_db()

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

Sauvegarde.

8.5.2.20 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

8.5.2.21 set_url_info()

```
def RIA_sql.C_sql.set_url_info (
    self,
    nom,
    date,
    taille,
    scr )
```

Sauvegarde.

Paramètres

<i>nom</i>	
<i>date</i>	
<i>taille</i>	
<i>scr</i>	

8.5.2.22 write_certfr_cve()

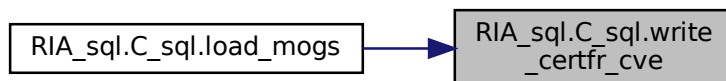
```
def RIA_sql.C_sql.write_certfr_cve (
    self,
    certfr,
    cve )
```

Ecrit dans la table CERTFR_tmp calcul le CRC automatiquement.

Paramètres

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

Voici le graphe des appelants de cette fonction :



8.5.2.23 write_certfr_tmp()

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

Ecrit dans la table CERTFR_tmp.

Paramètres

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

8.5.2.24 write_cpe_tmp()

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

Ecrit dans la table CVE_cpe_tmp.

Paramètres

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

8.5.2.25 write_cve_tmp()

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

Ecrit dans la table CVE_tmp.

Paramètres

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

8.5.2.26 write_sc()

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

Execute un script sqlite sans retour.

Paramètres

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

8.5.3 Documentation des données membres

8.5.3.1 Localdb

`RIA_sql.C_sql.Localdb`

le fichier MaBdd

8.5.3.2 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 9 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_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` (regex, obj)
Recherche regex pour Load_TAR_certfr.
- def `Load_TAR_certfr` (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.

8.6.2 Documentation des constructeurs et destructeur

8.6.2.1 `__init__()`

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

constructors

Paramètres

MaBdd	C_sql
-------	-------

8.6.3 Documentation des fonctions membres

8.6.3.1 Check_ALL_Wapper_Update()

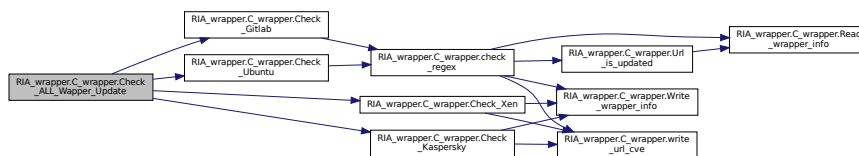
```
def RIA_wrapper.C_wrapper.Check_ALL_Wapper_Update (
    self,
    date )
```

Vérifie tous les editeurs en une seule fonction.

Paramètres

date	la date "YYYYMMDD" a verifier
------	-------------------------------

Voici le graphe d'appel pour cette fonction :

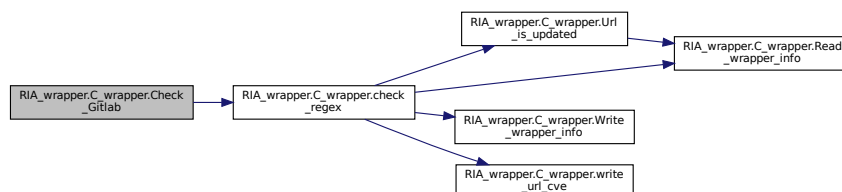


8.6.3.2 Check_Gitlab()

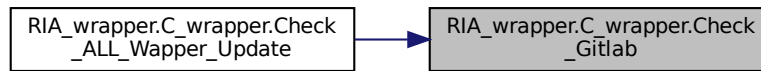
```
def RIA_wrapper.C_wrapper.Check_Gitlab (
    self )
```

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

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :

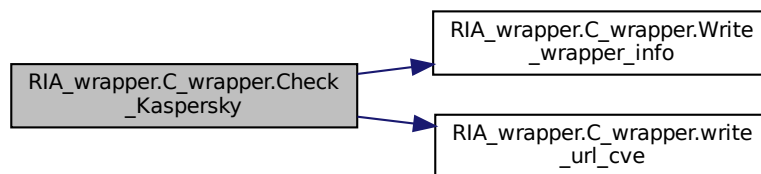


8.6.3.3 Check_Kaspersky()

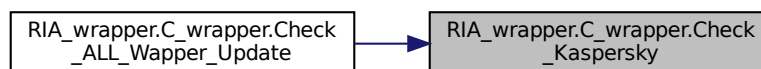
```
def RIA_wrapper.C_wrapper.Check_Kaspersky (
    self )
```

Verifie Kaspersky.

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.6.3.4 check_regex()

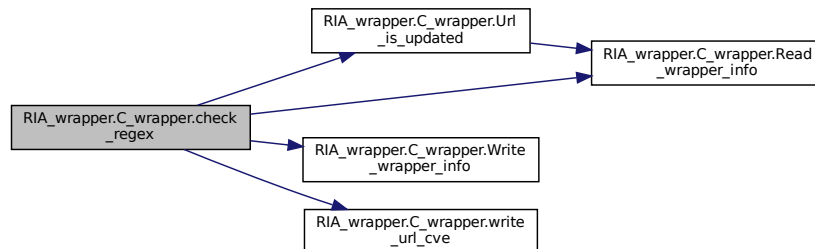
```
def RIA_wrapper.C_wrapper.check_regex (
    self,
    info )
```

Parse une url en regex.

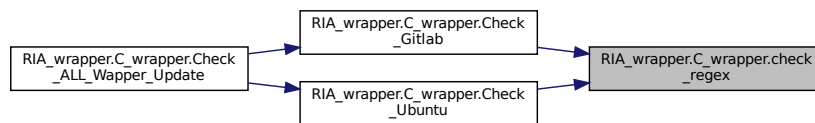
Paramètres

info	Un C_wrapper_info
------	-----------------------------------

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :

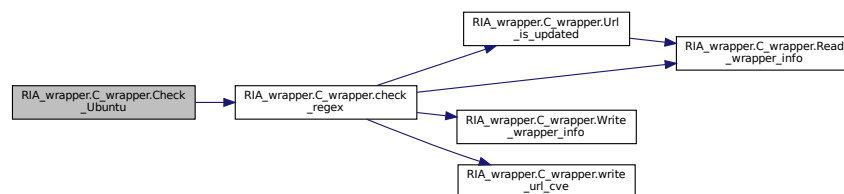


8.6.3.5 Check_Ubuntu()

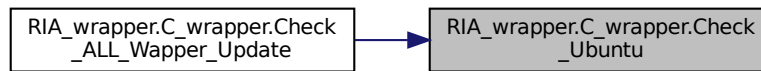
```
def RIA_wrapper.C_wrapper.Check_Ubuntu (
    self )
```

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 :

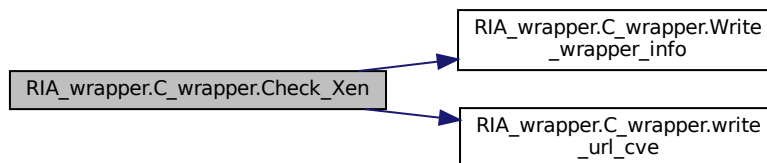


8.6.3.6 Check_Xen()

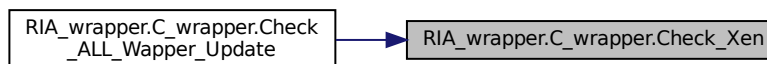
```
def RIA_wrapper.C_wrapper.Check_Xen (  
    self )
```

Verifie Xen.

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.6.3.7 Download_Certfr()

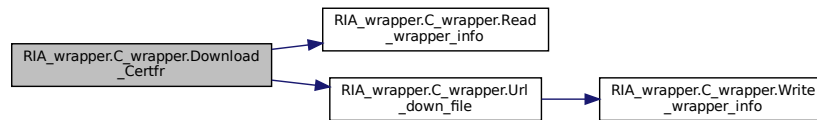
```
def RIA_wrapper.C_wrapper.Download_Certfr (  
    self,  
    EndDate )
```

Télécharge les Tar CERTFR si plus récent.

Paramètres

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

Voici le graphe d'appel pour cette fonction :

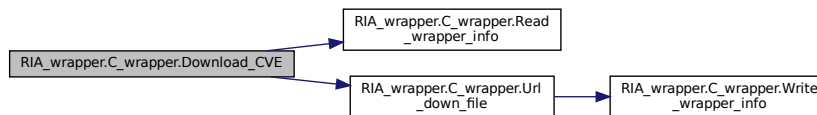


8.6.3.8 Download_CVE()

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

Télécharge les ZIP CVE NIST si plus récent.

Voici le graphe d'appel pour cette fonction :



8.6.3.9 Flush_cve()

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

Sauvegarde tous les couples Url/CVE trouvés.

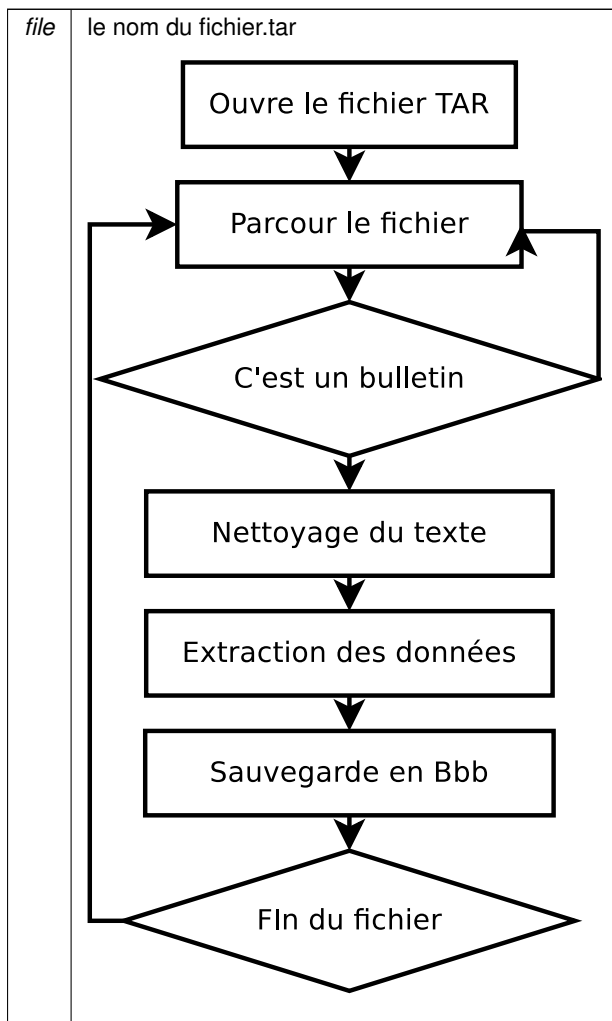
A faire traite les New

8.6.3.10 Load_TAR_certfr()

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

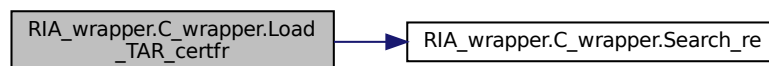
extrait les info CERTFR d'un TAR

Paramètres



A faire nettoyer le code

Voici le graphe d'appel pour cette fonction :



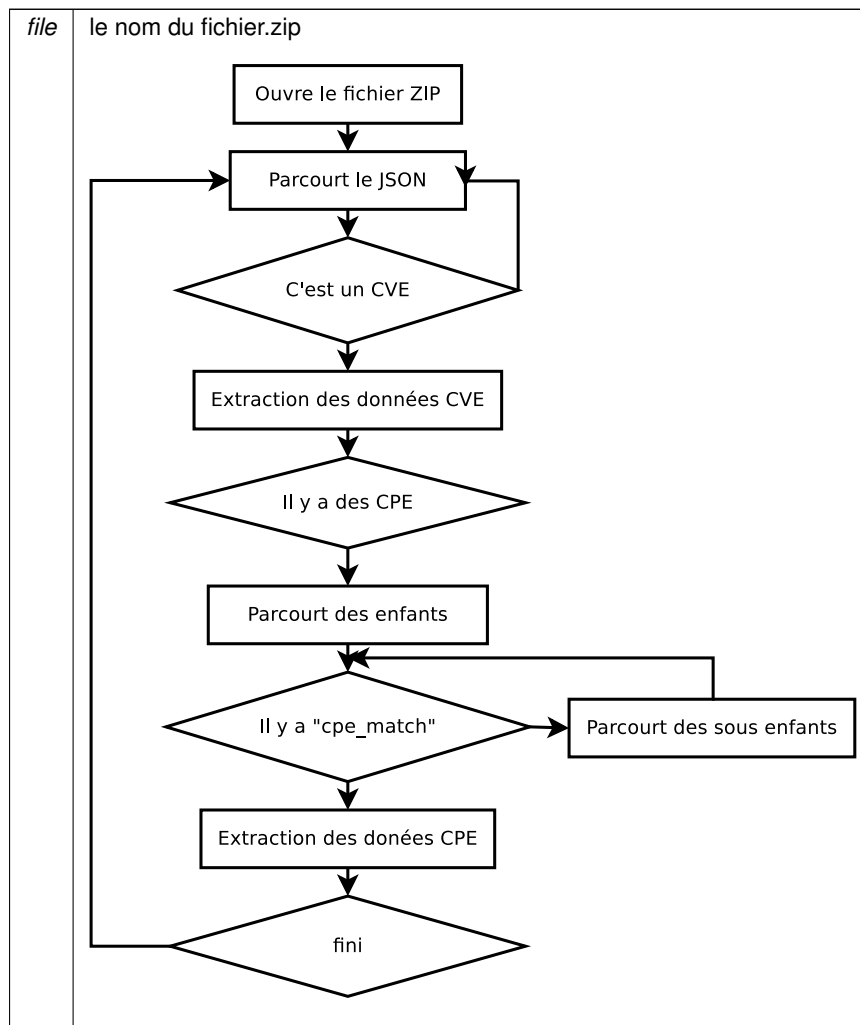
8.6.3.11 Load_ZIP_cve()

```

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

extraite les info CVE d'un zip

Paramètres



A faire nettoyer le code

8.6.3.12 `Read_wrapper_info()`

```

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

```

Lit en BDD des objet `C_wrapper_info`.

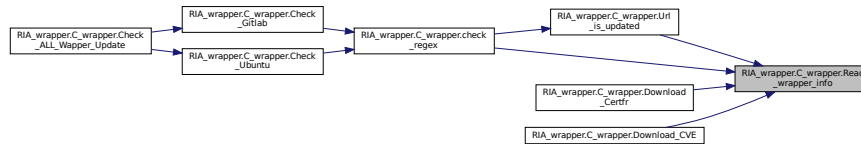
Paramètres

<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](#)

Voici le graphe des appelants de cette fonction :

**8.6.3.13 Reset_New()**

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

remet a 0 le champ New

8.6.3.14 Search_re()

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

Recherche regex pour Load_TAR_certfr.

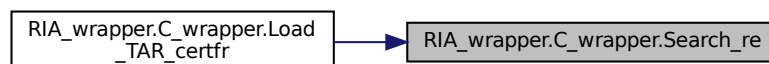
Paramètres

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

Renvoie

la string ou "

Voici le graphe des appelants de cette fonction :



8.6.3.15 Url_down_file()

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

Télécharge un fichiers si plus récent.

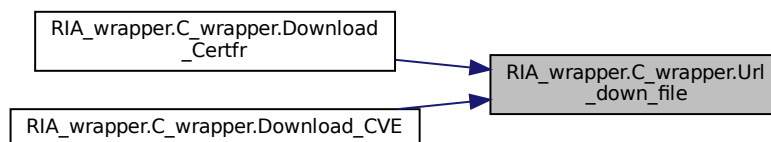
Paramètres

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

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



8.6.3.16 Url_is_updated()

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

Verifie si le header de la page distante est plus recent.

Paramètres

<i>info</i>	le C_wrapper_info a verifier
-------------	--

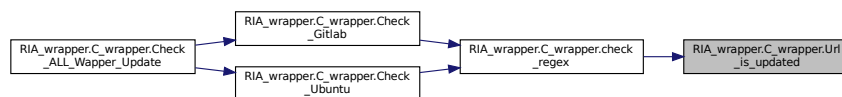
Renvoie

Boolean

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :

**8.6.3.17 write_url_cve()**

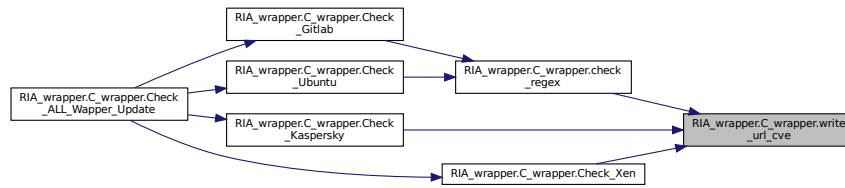
```
def RIA_wrapper.C_wrapper.write_url_cve (
    self,
    info )
```

Sauvegarde dans URL_ck.

Paramètres

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

Voici le graphe des appelants de cette fonction :



8.6.3.18 Write_wrapper_info()

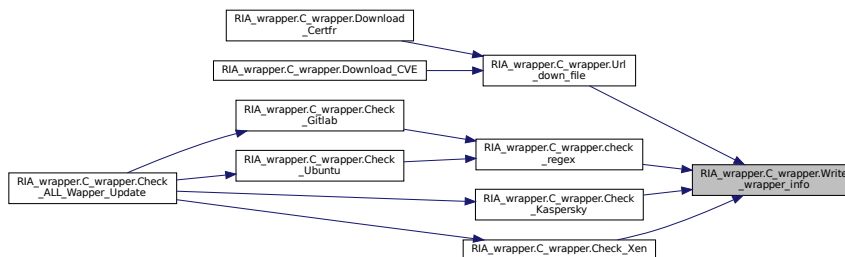
```
def RIA_wrapper.C_wrapper.Write_wrapper_info (
    self,
    info )
```

Sauvegarde en BDD un objet `C_wrapper_info`.

Paramètres

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

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
<ul style="list-style-type: none">+ __init__()

Fonctions membres publiques

— def [__init__](#)(self)

Attributs publics

— [Url](#)
— [Fichier](#)
— [Taille](#)
— [Rep](#)
— [Date](#)
— [Regex](#)
— [S_Url](#)
— [Module](#)
— [New](#)

8.7.1 Description détaillée

Class Objet info Wrapper.

8.7.2 Documentation des constructeurs et destructeur

8.7.2.1 `__init__()`

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

8.7.3 Documentation des données membres

8.7.3.1 Date

```
RIA_wrapper.C_wrapper_info.Date
```

8.7.3.2 Fichier

```
RIA_wrapper.C_wrapper_info.Fichier
```

8.7.3.3 Module

```
RIA_wrapper.C_wrapper_info.Module
```

8.7.3.4 New

```
RIA_wrapper.C_wrapper_info.New
```

8.7.3.5 Regex

```
RIA_wrapper.C_wrapper_info.Regex
```

8.7.3.6 Rep

```
RIA_wrapper.C_wrapper_info.Rep
```

8.7.3.7 S_Url

`RIA_wrapper.C_wrapper_info.S_Url`

8.7.3.8 Taille

`RIA_wrapper.C_wrapper_info.Taille`

8.7.3.9 Url

`RIA_wrapper.C_wrapper_info.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

Mon IA.

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

Mon IA.

Auteur

Frack113

Date

01/04/2020 Recherche d'Information Automatisée

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

01/04/2020 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

01/04/2020 Recherche d'Information Automatisée

Avertissement

la clé API est dans RIA_mskb.key

A faire optimiser le code

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

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

06/04/2020 module pour les recherches Internet

A faire traiter les New

Index

- `__init__`
 - `RIA_class.C_certfr`, [20](#)
 - `RIA_class.C_cpe`, [24](#)
 - `RIA_class.C_cve`, [28](#)
 - `RIA_mskb.C_mskb`, [32](#)
 - `RIA_sql.C_sql`, [38](#)
 - `RIA_wrapper.C_wrapper`, [48](#)
 - `RIA_wrapper.C_wrapper_info`, [60](#)
- `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`, [13](#)
- `Check_ALL_Wapper_Update`
 - `RIA_wrapper.C_wrapper`, [49](#)
- `Check_Gitlab`
 - `RIA_wrapper.C_wrapper`, [49](#)
- `Check_Kaspersky`
 - `RIA_wrapper.C_wrapper`, [50](#)
- `Check_Mskb_Update`
 - `RIA_mskb.C_mskb`, [32](#)
- `check_regex`
 - `RIA_wrapper.C_wrapper`, [50](#)
- `Check_Ubuntu`
 - `RIA_wrapper.C_wrapper`, [51](#)
- `Check_Xen`
 - `RIA_wrapper.C_wrapper`, [52](#)
- `clean_new`
 - `RIA_sql.C_sql`, [38](#)
- `clean_tmp`
 - `RIA_sql.C_sql`, [39](#)
- `close_db`
 - `RIA_sql.C_sql`, [39](#)
- `conf`
 - `RIA_class.C_cpe`, [25](#)
- `cpe23Uri`
 - `RIA_class.C_cpe`, [25](#)
- `crc`
 - `RIA_class.C_certfr`, [22](#)
 - `RIA_class.C_cpe`, [25](#)
 - `RIA_class.C_cve`, [29](#)
- `credit`
 - `RIA`, [14](#)
- `cve`
 - `RIA_class.C_cpe`, [25](#)
- `cvssV2`
 - `RIA_class.C_cve`, [29](#)
- `cvssV2base`
 - `RIA_class.C_cve`, [29](#)
- `cvssV3`
 - `RIA_class.C_cve`, [29](#)
- `cvssV3base`
 - `RIA_class.C_cve`, [30](#)
- `Date`
 - `RIA_wrapper.C_wrapper_info`, [61](#)
- `dateOrigine`
 - `RIA_class.C_certfr`, [22](#)
 - `RIA_class.C_cve`, [30](#)
- `dateUpdate`
 - `RIA_class.C_certfr`, [22](#)
 - `RIA_class.C_cve`, [30](#)
- `decode_file`
 - `RIA_class.C_certfr`, [20](#)
- `decode_link`
 - `RIA_class.C_certfr`, [21](#)
- `Download_Certfr`
 - `RIA_wrapper.C_wrapper`, [52](#)
- `Download_CVE`
 - `RIA_wrapper.C_wrapper`, [53](#)
- `encode_file`
 - `RIA_class.C_certfr`, [21](#)
- `encode_link`
 - `RIA_class.C_certfr`, [21](#)
- `Fichier`
 - `RIA_wrapper.C_wrapper_info`, [61](#)
- `file`
 - `RIA_class.C_certfr`, [22](#)
- `Flush_cve`
 - `RIA_wrapper.C_wrapper`, [53](#)
- `flush_tmp`
 - `RIA_sql.C_sql`, [39](#)
- `get_all_certfr_by_cve`
 - `RIA_sql.C_sql`, [39](#)
- `get_all_cpe_certfr`
 - `RIA_sql.C_sql`, [39](#)
- `get_all_cpe_uri`
 - `RIA_sql.C_sql`, [40](#)
- `get_all_cve_certfr`
 - `RIA_sql.C_sql`, [40](#)
- `get_all_new_certfr`
 - `RIA_sql.C_sql`, [40](#)

- get_all_new_url
 - RIA_sql.C_sql, 41
- get_all_orphan
 - RIA_sql.C_sql, 41
- get_certfr
 - RIA_sql.C_sql, 41
- get_info_certfr
 - RIA_mskb.C_mskb, 33
- get_Info_date
 - RIA_sql.C_sql, 42
- get_max_lg_uri_cpe
 - RIA_sql.C_sql, 42
- get_orphan_by_obj
 - RIA_sql.C_sql, 42
- get_sc
 - RIA_sql.C_sql, 43
- get_url_info
 - RIA_sql.C_sql, 43
- header
 - RIA_mskb.C_mskb, 36
- id
 - RIA_class.C_cpe, 26
 - RIA_class.C_cve, 30
- link
 - RIA_class.C_certfr, 22
- load_mogs
 - RIA_sql.C_sql, 43
- Load_TAR_certfr
 - RIA_wrapper.C_wrapper, 53
- Load_ZIP_cve
 - RIA_wrapper.C_wrapper, 54
- Localdb
 - RIA_sql.C_sql, 46
- MaBdd
 - RIA_mskb.C_mskb, 36
 - RIA_wrapper.C_wrapper, 59
- Module
 - RIA_wrapper.C_wrapper_info, 61
- mon_script
 - RIA, 14
- moncur
 - RIA_sql.C_sql, 47
- MS_to_STR
 - RIA, 14
- New
 - RIA_class.C_certfr, 22
 - RIA_class.C_cpe, 26
 - RIA_class.C_cve, 30
 - RIA_wrapper.C_wrapper_info, 61
- nom
 - RIA_class.C_certfr, 23
- obj
 - RIA_class.C_certfr, 23
- opérateur
 - RIA_class.C_cpe, 26
- Read_wrapper_info
 - RIA_wrapper.C_wrapper, 55
- README.md, 63
- Regex
 - RIA_wrapper.C_wrapper_info, 61
- Rep
 - RIA_wrapper.C_wrapper_info, 61
- reset
 - RIA_class.C_certfr, 21
 - RIA_class.C_cpe, 24
 - RIA_class.C_cve, 28
- reset_db
 - RIA_mskb.C_mskb, 33
- Reset_New
 - RIA_wrapper.C_wrapper, 56
- RIA, 13
 - CERT_to_STR, 13
 - credit, 14
 - mon_script, 14
 - MS_to_STR, 14
 - URI_to_FILE, 15
 - Write_CERTFR, 16
- RIA.py, 63
- RIA_class, 17
- RIA_class.C_certfr, 19
 - __init__, 20
 - crc, 22
 - dateOrigine, 22
 - dateUpdate, 22
 - decode_file, 20
 - decode_link, 21
 - encode_file, 21
 - encode_link, 21
 - file, 22
 - link, 22
 - New, 22
 - nom, 23
 - obj, 23
 - reset, 21
 - set_crc, 21
- RIA_class.C_cpe, 23
 - __init__, 24
 - conf, 25
 - cpe23Uri, 25
 - crc, 25
 - cve, 25
 - id, 26
 - New, 26
 - opérateur, 26
 - reset, 24
 - set_crc, 25
 - versionEndExcluding, 26
 - versionEndIncluding, 26
 - versionStartExcluding, 26
 - versionStartIncluding, 27
 - vulnerable, 27
- RIA_class.C_cve, 27

- [__init__, 28](#)
 - [crc, 29](#)
 - [cvssV2, 29](#)
 - [cvssV2base, 29](#)
 - [cvssV3, 29](#)
 - [cvssV3base, 30](#)
 - [dateOrigine, 30](#)
 - [dateUpdate, 30](#)
 - [id, 30](#)
 - [New, 30](#)
 - [reset, 28](#)
 - [set_crc, 29](#)
- [RIA_class.py, 64](#)
- [RIA_mskb, 17](#)
- [RIA_mskb.C_mskb, 31](#)
 - [__init__, 32](#)
 - [api_key, 36](#)
 - [api_type, 36](#)
 - [api_url, 36](#)
 - [Check_Mskb_Update, 32](#)
 - [get_info_certfr, 33](#)
 - [header, 36](#)
 - [MaBdd, 36](#)
 - [reset_db, 33](#)
 - [update_all_info, 33](#)
 - [update_all_url, 34](#)
 - [write_cve_kb, 34](#)
 - [write_product, 35](#)
- [RIA_mskb.py, 64](#)
- [RIA_sql, 17](#)
- [RIA_sql.C_sql, 37](#)
 - [__init__, 38](#)
 - [clean_new, 38](#)
 - [clean_tmp, 39](#)
 - [close_db, 39](#)
 - [flush_tmp, 39](#)
 - [get_all_certfr_by_cve, 39](#)
 - [get_all_cpe_certfr, 39](#)
 - [get_all_cpe_uri, 40](#)
 - [get_all_cve_certfr, 40](#)
 - [get_all_new_certfr, 40](#)
 - [get_all_new_url, 41](#)
 - [get_all_orphan, 41](#)
 - [get_certfr, 41](#)
 - [get_Info_date, 42](#)
 - [get_max_lg_uri_cpe, 42](#)
 - [get_orphan_by_obj, 42](#)
 - [get_sc, 43](#)
 - [get_url_info, 43](#)
 - [load_mogs, 43](#)
 - [Localdb, 46](#)
 - [moncur, 47](#)
 - [save_db, 44](#)
 - [set_Info_date, 44](#)
 - [set_url_info, 44](#)
 - [write_certfr_cve, 45](#)
 - [write_certfr_tmp, 45](#)
 - [write_cpe_tmp, 46](#)
 - [write_cve_tmp, 46](#)
 - [write_sc, 46](#)
- [RIA_sql.py, 65](#)
- [RIA_wrapper, 17](#)
- [RIA_wrapper.C_wrapper, 47](#)
 - [__init__, 48](#)
 - [Check_ALL_Wapper_Update, 49](#)
 - [Check_Gitlab, 49](#)
 - [Check_Kaspersky, 50](#)
 - [check_regex, 50](#)
 - [Check_Ubuntu, 51](#)
 - [Check_Xen, 52](#)
 - [Download_Certfr, 52](#)
 - [Download_CVE, 53](#)
 - [Flush_cve, 53](#)
 - [Load_TAR_certfr, 53](#)
 - [Load_ZIP_cve, 54](#)
 - [MaBdd, 59](#)
 - [Read_wrapper_info, 55](#)
 - [Reset_New, 56](#)
 - [Search_re, 56](#)
 - [Url_down_file, 57](#)
 - [Url_is_updated, 57](#)
 - [write_url_cve, 58](#)
 - [Write_wrapper_info, 59](#)
- [RIA_wrapper.C_wrapper_info, 60](#)
 - [__init__, 60](#)
 - [Date, 61](#)
 - [Fichier, 61](#)
 - [Module, 61](#)
 - [New, 61](#)
 - [Regex, 61](#)
 - [Rep, 61](#)
 - [S_Url, 61](#)
 - [Taille, 62](#)
 - [Url, 62](#)
- [RIA_wrapper.py, 65](#)
- [S_Url](#)
 - [RIA_wrapper.C_wrapper_info, 61](#)
- [save_db](#)
 - [RIA_sql.C_sql, 44](#)
- [Search_re](#)
 - [RIA_wrapper.C_wrapper, 56](#)
- [set_crc](#)
 - [RIA_class.C_certfr, 21](#)
 - [RIA_class.C_cpe, 25](#)
 - [RIA_class.C_cve, 29](#)
- [set_Info_date](#)
 - [RIA_sql.C_sql, 44](#)
- [set_url_info](#)
 - [RIA_sql.C_sql, 44](#)
- [Taille](#)
 - [RIA_wrapper.C_wrapper_info, 62](#)
- [update_all_info](#)
 - [RIA_mskb.C_mskb, 33](#)
- [update_all_url](#)

- RIA_mskb.C_mskb, [34](#)
- URI_to_FILE
 - RIA, [15](#)
- Url
 - RIA_wrapper.C_wrapper_info, [62](#)
- Url_down_file
 - RIA_wrapper.C_wrapper, [57](#)
- Url_is_updated
 - RIA_wrapper.C_wrapper, [57](#)
- versionEndExcluding
 - RIA_class.C_cpe, [26](#)
- versionEndIncluding
 - RIA_class.C_cpe, [26](#)
- versionStartExcluding
 - RIA_class.C_cpe, [26](#)
- versionStartIncluding
 - RIA_class.C_cpe, [27](#)
- vulnerable
 - RIA_class.C_cpe, [27](#)
- Write_CERTFR
 - RIA, [16](#)
- write_certfr_cve
 - RIA_sql.C_sql, [45](#)
- write_certfr_tmp
 - RIA_sql.C_sql, [45](#)
- write_cpe_tmp
 - RIA_sql.C_sql, [46](#)
- write_cve_kb
 - RIA_mskb.C_mskb, [34](#)
- write_cve_tmp
 - RIA_sql.C_sql, [46](#)
- write_product
 - RIA_mskb.C_mskb, [35](#)
- write_sc
 - RIA_sql.C_sql, [46](#)
- write_url_cve
 - RIA_wrapper.C_wrapper, [58](#)
- Write_wrapper_info
 - RIA_wrapper.C_wrapper, [59](#)