

# Solução do desafio Clavis

01/08/2016

Como encontrei uma boa biblioteca para a API charts do google no python e a do perl não funcionou, vou desenvolver em python.

Vamos chamar a aplicação de chart\_CVE.py

O objetivo é receber uma lista de CVEs e buscar as informações com GETs em [https://web.nvd.nist.gov/view/vuln/detail?vulnId=\\*](https://web.nvd.nist.gov/view/vuln/detail?vulnId=*). Depois de parsear a pagina, vou armazenar as informacoes em um array.

```
python chart_CVE.py lista_CVE.txt
```

O arquivo TXT com a lista de CVE tera o seguinte formato (um CVE em cada linha):

```
CVE-2016-5511
CVE-2016-5512
CVE-2016-5513
CVE-2016-5514
CVE-2016-5515
```

Como output, o programa ira gerar 4 arquivos graficos:

- chartV2.png
- chartV3.png
- barV2.png
- barV3.png

E um arquivo contendo a tabela de CVEs contendo os seguintes dados:

- tabela.csv

```
CVE_NUMBER, CVSSV2_SCORE, CVSSV2_IMPACT, CSSV2_EXPLOITABILITY, CVSSV3_SCORE, CVSSV3_IMPACT,
CSSV3_EXPLOITABILITY
```

## Instalação

Assumo que o python versão 2.6 ou 2.7 está instalado, e o instalador de pacotes do python, *pip*, também está. Para executar o request GET usarei a *urllib2* do python, ela é padrão na versão 2.X.

Vamos instalar o modulo para fazer o parse na página, este é um módulo que já usei para fazer *crawlers*:

```
pip install bs4
```

E o modulo para acessar a API do google (<https://github.com/gak/pygooglechart/tree/master/examples>). Eu rodei os exemplos deste módulo e funcionaram perfeitamente.

```
pip install pygooglechart
```

Fuçando até achei uma lista de CVEs para download aqui: <https://cve.mitre.org/data/downloads/allitems.csv> Baixei e gerei uma lista de teste com os últimos 100 items:

```
cut -d, -f1 < allitems.csv | tail -100 >lista_CVE.txt
```



Realmente não avancei muito no meu script, so cheguei até ler o arquivo. Mas testei e gerei charts com a api do google. Segue o script incompleto.

```
1"""
2
3 Desafio clavis
4
5
6 """
7
8 import sys
9 from bs4 import BeautifulSoup
10 import urllib2
11 import re
12
13 filename = sys.argv[1]
14
15 file = open(filename, 'r')
16
17 cve_list=[]
18 for line in file:
19     cve_list.append(line.rstrip())
20
21
22 file.close()
23
24 print cve_list
25
26
27 # executa o GET
28
29
30 # parse das informacoes na pagina
31
32
33 # armazena em uma lista
34
35
36 # gera graficos
-----
```

Nota, esta documentação foi gerada em ASCIIDOC

Andre Oliveira Dias <aod7br@gmail.com>