# ANALISADOR PORTÁTIL DE PENDRIVES COM RASPBERRY PI 4 B: UMA SOLUÇÃO EFICIENTE PARA DETECÇÃO DE AMEAÇAS EM MEMÓRIAS PORTÁTEIS

Ten JEFFERSON **ADINIZ** BORGES FERREIRA
Sgt ANDERSON **LUCIO** GOMES

**Resumo:** Este trabalho propõe o uso do Raspberry Pi 4 B como base para a construção de um analisador portátil de pendrives, oferecendo uma solução eficiente para a detecção de ameaças em dispositivos de armazenamento portáteis. Utilizando a API do VirusTotal, o Raspberry Pi 4 B, um computador de placa única acessível e versátil, possibilita a monitorização e análise eficaz de dados em dispositivos USB, identificando potenciais ameaças e fortalecendo a segurança da informação. A integração com a API do VirusTotal adiciona uma camada extra de proteção, permitindo a verificação de arquivos contra uma vasta base de dados de malwares conhecidos. Neste artigo, serão exploradas as funcionalidades principais do Raspberry Pi 4 B, discutidas suas vantagens, e apresentados exemplos práticos de sua aplicação na análise de pendrives, ressaltando sua importância como um ativo valioso em infraestruturas de segurança da informação.

**Palavras-Chave:** Raspberry PI, Pen Drives, Vírus Total

## 1. INTRODUÇÃO

Na era digital, em que a troca rápida e segura de informações é crucial, a proteção de dados se torna uma preocupação central. Dispositivos de armazenamento portáteis, como pendrives, são amplamente utilizados para transferir dados entre diferentes sistemas. No entanto, essa conveniência está associada a riscos significativos, uma vez que pendrives podem se tornar veículos de disseminação de malwares e outras ameaças cibernéticas.

A detecção de ameaças em memórias portáteis é, portanto, uma medida de segurança indispensável para proteger sistemas e redes de possíveis ataques. Tanto organizações quanto indivíduos necessitam de ferramentas eficazes que permitam a análise rápida desses dispositivos, garantindo que não representem um risco. Nesse contexto, o Raspberry Pi 4 B emerge como uma solução acessível e eficiente. O Raspberry Pi 4 B é um computador de placa única, desenvolvido pela Raspberry Pi Foundation, reconhecido por sua versatilidade e baixo custo, pode ser utilizado como base para a construção de um analisador portátil de pendrives.


A integração da API do VirusTotal ao Raspberry Pi 4 B adiciona uma poderosa camada de segurança. O VirusTotal é um serviço que agrega resultados de múltiplos

mecanismos antivírus e ferramentas de análise de malware, possibilitando uma verificação abrangente de arquivos suspeitos. Ao utilizar essa API, é possível escanear arquivos armazenados em pendrives contra uma extensa base de dados de malwares conhecidos, ampliando significativamente a capacidade de detecção e resposta a ameaças.

Este artigo explora a importância da proteção de dados e sistemas contra as ameaças introduzidas por memórias portáteis. Discutiremos como o Raspberry Pi 4 B, em conjunto com a API do VirusTotal, pode ser utilizado para criar um analisador portátil de pendrives. Além disso, serão apresentadas as vantagens dessa solução, bem como exemplos práticos de sua aplicação, ressaltando sua relevância como um componente crucial em infraestruturas de segurança da informação.

## 2. DESENVOLVIMENTO

### 2.1 ARQUITETURA E CONFIGURAÇÃO DO ANALISADOR DE PENDRIVES COM Raspberry pi 4 B

A configuração do Raspberry pi 4 B como um analisador de pendrives começa com a escolha dos componentes de hardware e software. O Raspberry pi 4 B, com sua combinação de tamanho compacto e poder de processamento suficiente, é ideal para esta tarefa. Neste subitem, discutiremos:

Seleção de Hardware as características principais do Raspberry Pi 4 incluem:

- **Processador:** Broadcom BCM2711, um SoC (System on a Chip) que possui uma CPU quad-core ARM Cortex-A72 de 64 bits, com clock de 1,5 GHz.
- **Memória:** Disponível em versões com 2 GB, 4 GB ou 8 GB de memória RAM LPDDR4.
- **Armazenamento:** Utiliza cartões microSD para armazenamento principal, mas também suporta boot por USB, permitindo o uso de SSDs externos para melhorar a performance.
- **Conectividade:** Possui portas USB 3.0 e USB 2.0, Gigabit Ethernet, e conectividade sem fio integrada, incluindo Wi-Fi 802.11ac de banda dupla e Bluetooth 5.0.
- **Vídeo e Gráficos:** Suporta saída de vídeo em 4K a 60 fps através de duas portas micro HDMI, sendo ideal para aplicações de multimídia.
- **Sistema Operacional:** Compatível com várias distribuições de sistemas operacionais baseados em Linux, como o Raspberry Pi OS, além de ser capaz de rodar sistemas operacionais como Ubuntu e até mesmo versões de Windows 10 IoT Core.

O Raspberry Pi 4 é especialmente valorizado por seu equilíbrio entre desempenho e custo, tornando-o uma escolha popular para projetos educacionais, domésticos, industriais e de pesquisa.

Além do Raspberry pi 4 B, é necessário um adaptador USB OTG para conectar os pendrives, uma fonte de alimentação estável, e um cartão microSD com uma capacidade mínima de 16 GB para armazenar o sistema operacional e os arquivos temporários. Dependendo do ambiente de uso, um case protetor pode ser recomendado para proteger o dispositivo durante o transporte.

Instalação do Sistema Operacional: O Raspberry Pi OS (anteriormente conhecido como Raspbian) é a escolha ideal devido à sua compatibilidade e suporte à comunidade. Discutiremos as etapas de instalação do sistema operacional, incluindo o download da imagem, o uso do Raspberry Pi Imager, e as primeiras configurações, como a habilitação do SSH para acesso remoto.

Alimentação: A alimentação do dispositivo será feita por duas baterias 18650 de 2A conectadas em série. Essa configuração fornece uma tensão nominal de 7.4V, o que é adequado para alimentar o Raspberry pi 4 B por um período prolongado. Nota: O tempo de utilização estimado é baseado em uma carga total das baterias e em condições de operação típicas. O uso de periféricos adicionais ou uma carga parcial das baterias pode reduzir esse tempo.

Configuração de Rede: Como a API do VirusTotal requer conexão à internet, abordaremos a configuração da conectividade via Wi-Fi, com foco na conexão ao roteador Wi-Fi do smartphone do usuário. **2.2 INTEGRAÇÃO COM A API DO VIRUSTOTAL**

A integração com a API do VirusTotal é uma das funcionalidades mais poderosas do analisador de pendrives, permitindo uma verificação eficaz de arquivos em busca de ameaças. Este subitem detalha cada etapa do processo:

**Registro e Configuração da API:**

O primeiro passo para utilizar a API do VirusTotal é o registro no serviço para obtenção de uma chave de API, que será usada para autenticação nas chamadas. Vamos explicar como configurar essa chave no Raspberry Pi, sugerindo o uso de variáveis de ambiente para armazená-la de forma segura, facilitando seu acesso nos scripts e mantendo a segurança dos dados.

**Criação de Scripts para Chamadas à API:**

Usando Python, vamos demonstrar como criar scripts que automatizam o envio de arquivos armazenados em pendrives para análise no VirusTotal. Serão discutidas as bibliotecas essenciais, como requests, e serão fornecidos exemplos práticos de código que realizam chamadas à API, obtêm relatórios de verificação e processam as respostas, permitindo a identificação de ameaças de maneira eficiente. O foco será em garantir que o processo seja ágil e possa ser facilmente integrado em soluções de segurança maiores.

**Limitações da API e Soluções Alternativas:** Como a API do VirusTotal impõe limites no número de chamadas por minuto, apresentaremos estratégias para otimizar seu uso. Entre elas, destacamos a verificação seletiva de arquivos com base no hash (por exemplo, SHA-256) para evitar a análise redundante de arquivos já verificados, e o uso de cache local para armazenar resultados temporários. Além disso, alternativas viáveis, como outras APIs de antivírus gratuitas ou de código aberto, que possam complementar o VirusTotal, oferecendo maior flexibilidade na detecção de ameaças e ampliando a cobertura de segurança.

## 2.3 PROCESSAMENTO E ANÁLISE DE ARQUIVOS

O processamento e a análise dos arquivos são fundamentais para a eficácia do analisador. Neste subitem, abordaremos:

Análise Preliminar de Arquivos: Antes de enviar arquivos para o VirusTotal, o Raspberry Pi pode realizar uma análise preliminar. Discutiremos técnicas para verificar a integridade dos arquivos, identificar extensões suspeitas, e analisar metadados que possam indicar comportamento malicioso, como a presença de scripts autorun.

Uso de Bancos de Dados Locais: Para acelerar o processo de análise, o Raspberry Pi pode utilizar bancos de dados locais de hashes de malwares conhecidos. Explicaremos como atualizar e manter esses bancos de dados, além de como usá-los para comparações rápidas antes de recorrer à API do VirusTotal.

Análise Comportamental e de Padrões: Para arquivos que não correspondem a malwares conhecidos, discutiremos como implementar técnicas de análise comportamental. Isso pode incluir a execução de scripts em um ambiente sandbox no Raspberry Pi para observar comportamentos anômalos, como tentativas de se conectar à internet ou modificar arquivos do sistema.

Alertas de Ameaças: Após a análise, o Raspberry Pi indicará a presença de ameaças por meio de um sistema de LEDs. Um LED verde acenderá para indicar que o pendrive está limpo, enquanto um LED vermelho indicará a detecção de uma ameaça. Abordaremos como configurar esses LEDs para reagirem em tempo real aos resultados da análise, garantindo uma sinalização clara e imediata do status do dispositivo.

## 2.4 IMPLEMENTAÇÃO E TESTES PRÁTICOS

A implementação prática da solução é crucial para validar sua eficácia. Este subitem está focado em:

Controlar os LEDs conectados ao Raspberry Pi durante o processo de verificação de pendrives, você pode usar a biblioteca **GPIO**

do Python, que permite controlar os pinos GPIO para ligar e desligar os LEDs.

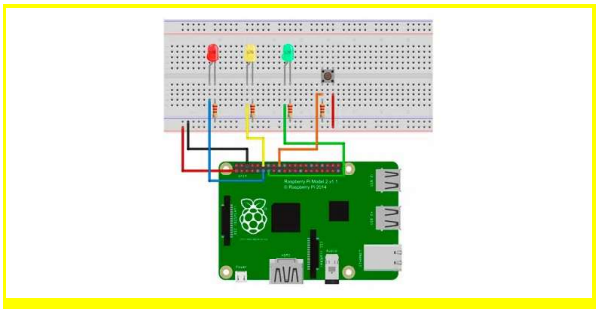
Os LEDs estão conectados aos pinos GPIO do Raspberry Pi funcionam da seguinte maneira:

O **LED amarelo** acende durante a verificação.

Quando a verificação terminar:

O **LED vermelho** acende se houver uma ameaça.

O **LED verde** acende se o pendrive estiver seguro.



O passo a passo da implementação é o seguinte:

**Atualize o sistema:**

```bash
sudo apt update
sudo apt upgrade
```

Instalar o Python e pip (caso não esteja instalado):

```bash
sudo apt install python3 python3-pip
```

**2. Obtenção da API Key do VirusTotal**

Para usar a API do VirusTotal, você precisa obter uma chave de API gratuita ou paga. Basta criar uma conta no VirusTotal, acessar o painel de controle e copiar sua chave de API.

**3. Instalar Bibliotecas Necessárias**

Para fazer chamadas HTTP e interagir com a API do VirusTotal, você usará a biblioteca requests. Instale-a usando o pip:

```bash
pip3 install requests
```

**4. Criar Script em Python para Chamadas à API**

Agora, você pode criar um script que envia arquivos de pendrives para análise no VirusTotal. Aqui está um exemplo de script básico:

```python
import os
import requests

# Sua chave de API do VirusTotal
API_KEY = 'SUA_CHAVE_API_AQUI'

# URL para enviar o arquivo à API do VirusTotal
url = 'https://www.virustotal.com/vtapi/v2/file/scan'

# Função para enviar o arquivo
def enviar_arquivo_virus_total(file_path):
    # Verifica se o arquivo existe
    if not os.path.isfile(file_path):
        print(f"Arquivo {file_path} não encontrado.")
        return

    # Dados para a API
    params = {'apikey': API_KEY}
    files = {'file': (file_path, open(file_path, 'rb'))}

    # Faz a chamada à API
    response = requests.post(url, files=files, params=params)

    # Verifica a resposta da API
    if response.status_code == 200:
        print(f"Arquivo enviado com sucesso. Resposta: {response.json()}")
    else:
        print(f"Erro ao enviar o arquivo. Código HTTP: {response.status_code}")

# Caminho do arquivo no pendrive para análise
caminho_do_arquivo = '/media/pi/pendrive/nome_do_arquivo.ext'

# Envia o arquivo para o VirusTotal
enviar_arquivo_virus_total(caminho_do_arquivo)
```

### 5. Verificar o Status da Análise

Depois de enviar o arquivo, você pode verificar o status da análise usando outro endpoint da API. Aqui está como fazer isso:

```python
import requests

# URL para verificar o relatório da API
url_report = 'https://www.virustotal.com/vtapi/v2/file/report'

# Função para obter o relatório de análise
def obter_relatorio_virus_total(resource_id):
    params = {'apikey': API_KEY, 'resource': resource_id}

    response = requests.get(url_report, params=params)

    if response.status_code == 200:
        json_response = response.json()
        print(f"Relatório recebido: {json_response}")
        # Processa os resultados conforme necessário
    else:
        print(f"Erro ao obter o relatório. Código HTTP: {response.status_code}")

# Resource ID ou SHA-256 do arquivo para buscar o relatório
resource_id = 'ID_DO_ARQUIVO_AQUI'
obter_relatorio_virus_total(resource_id)
```

### 6. Considerações sobre Limites da API

- A versão gratuita do VirusTotal permite até 4 solicitações de análise por minuto.
- Para evitar atingir o limite, você pode implementar verificação pelo hash do arquivo antes de fazer uma nova solicitação.
- Armazenar localmente os resultados (cache) também pode ajudar a evitar enviar o mesmo arquivo repetidamente.

## 7. Execução Automática no Raspberry Pi

O Raspberry Pi pode monitorar automaticamente os pendrives conectados e envie os arquivos para o VirusTotal, com um script que verifica novos dispositivos USB conectados e inicia o processo automaticamente.

```python
import os
import time

# Diretório onde os pendrives são montados no Raspberry Pi
usb_directory = '/media/pi/'

def verificar_dispositivos_usb():
    while True:
        dispositivos = os.listdir(usb_directory)
        if dispositivos:
            print(f"Dispositivo detectado: {dispositivos}")
            # Itera sobre os arquivos do pendrive e envia para análise
            for dispositivo in dispositivos:
                caminho_do_dispositivo = os.path.join(usb_directory, dispositivo)
                for root, dirs, files in os.walk(caminho_do_dispositivo):
                    for file in files:
                        file_path = os.path.join(root, file)
                        enviar_arquivo_virus_total(file_path)
        else:
            print("Nenhum pendrive detectado.")
        time.sleep(10)

# Verifica dispositivos USB constantemente
verificar_dispositivos_usb()
```

Os LEDs estão conectados aos pinos GPIO do Raspberry Pi. A ideia é o **LED amarelo** acende durante a verificação. Quando a verificação terminar:O **LED vermelho** acende se houver uma ameaça. O **LED verde** acende se o pendrive estiver seguro.

## Configurando os LEDs com GPIO:

1. **Instalar a biblioteca RPi.GPIO:** Se ainda não tiver a biblioteca instalada, instale-a com o seguinte comando:

```bash
sudo apt install python3-rpi.gpio
```

```python
import RPi.GPIO as GPIO
import time
import requests
import os

# Configuração dos pinos GPIO para os LEDs
LED_VERDE = 17   # Pino GPIO para o LED verde
LED_AMARELO = 27  # Pino GPIO para o LED amarelo
LED_VERMELHO = 22  # Pino GPIO para o LED vermelho

# Configuração inicial dos pinos GPIO
GPIO.setmode(GPIO.BCM)
GPIO.setup(LED_VERDE, GPIO.OUT)
GPIO.setup(LED_AMARELO, GPIO.OUT)
GPIO.setup(LED_VERMELHO, GPIO.OUT)

# Função para limpar os LEDs (desligar todos)
def limpar_leds():
    GPIO.output(LED_VERDE, GPIO.LOW)
    GPIO.output(LED_AMARELO, GPIO.LOW)
    GPIO.output(LED_VERMELHO, GPIO.LOW)

# Configuração da API VirusTotal
API_KEY = 'SUA_CHAVE_API_AQUI'
url = 'https://www.virustotal.com/vtapi/v2/file/scan'
url_report = 'https://www.virustotal.com/vtapi/v2/file/report'
```

```python
# Função para enviar o arquivo para o VirusTotal
def enviar_arquivo_virus_total(file_path):
    limpar_leds()
    print("Iniciando verificação...")

    # Acende o LED amarelo durante a verificação
    GPIO.output(LED_AMARELO, GPIO.HIGH)

    # Envia o arquivo para o VirusTotal
    if not os.path.isfile(file_path):
        print(f"Arquivo {file_path} não encontrado.")
        return

    params = {'apikey': API_KEY}
    files = {'file': (file_path, open(file_path, 'rb'))}
    response = requests.post(url, files=files, params=params)

    if response.status_code == 200:
        json_response = response.json()
        resource_id = json_response['resource']
        # Faz a verificação do relatório
        time.sleep(30)  # Espera 30 segundos para que o relatório seja gerado
        return obter_relatorio_virus_total(resource_id)
    else:
        print(f"Erro ao enviar o arquivo. Código HTTP: {response.status_code}")
```

```python
# Função para monitorar os dispositivos USB montados
def monitorar_pendrive():
    usb_directory = '/media/pi/'  # Diretório onde os pendrives são montados
    dispositivos_montados = set()

    while True:
        dispositivos_atual = set(os.listdir(usb_directory))

        # Detecta novos dispositivos conectados
        novos_dispositivos = dispositivos_atual - dispositivos_montados
        if novos_dispositivos:
            print(f"Novo pendrive detectado: {novos_dispositivos}")
            for dispositivo in novos_dispositivos:
                caminho_do_dispositivo = os.path.join(usb_directory, dispositivo)
                for root, dirs, files in os.walk(caminho_do_dispositivo):
                    for file in files:
                        file_path = os.path.join(root, file)
                        enviar_arquivo_virus_total(file_path)

        dispositivos_montados = dispositivos_atual

        # Verifica a cada 5 segundos por novos pendrives
        time.sleep(5)

# Função principal
try:
    limpar_leds()
    monitorar_pendrive()
except KeyboardInterrupt:
    print("Programa interrompido.")
finally:
    limpar_leds()
    GPIO.cleanup()
```

: A função monitorar_pendrive() verifica constantemente o diretório /media/pi/, onde os pendrives são montados automaticamente no Raspberry Pi. Quando detecta um novo dispositivo, inicia o processo de verificação dos arquivos contidos no pendrive.

**Verificação de Arquivos**: Para cada pendrive detectado, o código caminha pelas pastas e arquivos contidos nele e envia cada arquivo para a análise no VirusTotal.

**Controle dos LEDs**:

- O **LED amarelo** acende durante a verificação.
- Se for detectada uma ameaça, o **LED vermelho** acende.
- Se o pendrive estiver seguro, o **LED verde** será ativado.

**Loop de Monitoramento**: O loop verifica novos dispositivos a cada 5 segundos e atualiza a lista de dispositivos montados, para garantir que detecte pendrives recém-conectados.

**Executar o Script**: Salve o script em um arquivo Python, por exemplo verificador_pendrive.py, e execute no Raspberry Pi:

Conecte um pendrive e observe o comportamento dos LEDs. Durante a verificação, o LED amarelo acenderá, e quando o processo for concluído, o LED vermelho ou verde será ativado, dependendo dos resultados.

## 2.5 VANTAGENS E DESAFIOS DA SOLUÇÃO

Vantagens: Destacamos as principais vantagens do uso do Raspberry pi 4 B como analisador de pendrives, incluindo seu custo acessível, sua portabilidade, e a facilidade de configuração e uso, ele pode ser integrado em infraestruturas de segurança maiores e sua utilidade como ferramenta de resposta a incidentes.

Desafios Técnicos: as limitações do hardware, como a capacidade de processamento e armazenamento, e como isso pode afetar a análise de grandes volumes de dados ou de arquivos complexos. A conectividade de rede limitada em ambientes com pouca infraestrutura também será considerada.

Manutenção e Atualizações: Manter a eficácia da solução requer atualizações frequentes tanto do software quanto das bases de dados de malwares. As estratégias para garantir que o Raspberry Pi esteja sempre atualizado e preparado para detectar novas ameaças, incluindo a automação de atualizações e a verificação de integridade do sistema.

Escalabilidade e Adaptação: Finalmente, a solução pode ser escalada para uso em ambientes maiores ou adaptada para funções adicionais, como a análise de outros tipos de dispositivos USB, como discos rígidos externos ou smartphones.

## CONCLUSÃO

O uso do Raspberry pi 4 B como base para um analisador portátil de pendrives demonstra ser uma solução eficaz e acessível para a detecção de ameaças em dispositivos de armazenamento portáteis. Ao integrar a API do VirusTotal, o dispositivo é capaz de realizar verificações profundas e abrangentes, protegendo sistemas e redes contra malwares conhecidos. A configuração do dispositivo para se conectar ao Wi-Fi roteado do smartphone do usuário garante mobilidade e flexibilidade, permitindo que a análise de pendrives seja realizada em diversos ambientes sem a necessidade de infraestrutura adicional.

A simplicidade do sistema de alertas por LEDs, que indica a presença ou ausência de ameaças de maneira direta e visual, torna a solução prática e fácil de usar, tanto para profissionais de segurança da informação quanto para usuários menos experientes. A escolha de utilizar baterias 18650 em série para a alimentação do dispositivo também assegura um tempo de operação prolongado, fazendo com que o analisador seja confiável e eficiente mesmo em situações onde o acesso à energia elétrica é limitado.

Este projeto destaca a importância de soluções de segurança cibernética portáteis e acessíveis, especialmente em um cenário onde a mobilidade é cada vez mais valorizada.

Com as melhorias e adaptações discutidas ao longo deste artigo, o analisador de pendrives com Raspberry pi 4 B tem opotencial de se tornar uma ferramenta essencial em infraestruturas de segurança, fornecendo uma defesa robusta contra ameaças cibernéticas provenientes de dispositivos USB.

## REFERÊNCIAS BIBLIOGRÁFICAS


RASPBERRY PI FOUNDATION, **Getting started with your Raspberry Pi**. Disponível em:

RASPBERRY PI FOUNDATION. *Getting started with Raspberry Pi*. Disponível em: https://www.raspberrypi.com/documentation/computers/getting-started.html/. Acesso em: 26 ago. 2024.

PYPI. *virustotal-api 1.1.11*. Disponível em: https://pypi.org/project/virustotal-api/. Acesso em: 27 ago. 2024.

VIRUSTOTAL. *VirusTotal API v2.0 - Getting started*. Disponível em: https://docs.virustotal.com/v2.0/reference/getting-started. Acesso em: 27 ago. 2024. PYPI. *virustotal-python 0.1.2*. Disponível em: https://pypi.org/project/virustotal-python/. Acesso em: 27 ago. 2024.

FORUMS. *Raspberry Pi - Monitoring USB device connections*. Disponível em: https://forums.raspberrypi.com/viewtopic.php?t=318933. Acesso em: 27 ago. 2024.

**MAKERHERO.** *Projetos com Raspberry Pi*. Disponível em: https://www.makerhero.com/blog/projetos-com-raspberry-pi/. Acesso em: 4 set. 2024.

**SILVA, João P.** *Detecção de Ameaças Cibernéticas em Dispositivos USB*. 2022. Dissertação (Mestrado em Segurança da Informação) – Universidade Federal de São Paulo, São Paulo, 2022.


**Anexo A** - Sequência de instalação do Virus Total no

```
$ pip install virustotal-api

from __future__ import print_function
import json
import hashlib
from virus_total_apis import PublicApi as
VirusTotalPublicApi

API_KEY = 'Sign-Up for API Key at
virustotal.com'

EICAR =
"X5O!P%@AP[4\PZX54(P^)7CC)7}$EIC
AR-STANDARD-ANTIVIRUS-TEST-FILE!
$H+H*".encode('utf-8')
EICAR_MD5 =
hashlib.md5(EICAR).hexdigest()

vt = VirusTotalPublicApi(API_KEY)

response =
vt.get_file_report(EICAR_MD5)
print(json.dumps(response,
sort_keys=False, indent=4))

{
   "response_code": 200,
   "results": {
      "scan_id":
"275a021bbfb6489e54d471899f7db9d16
63fc695ec2fe2a2c4538aabf651fd0f-1397
510237",
```

```
    "sha1":
"3395856ce81f2b7382dee72602f798b64
2f14140",
    "resource":
"44d88612fea8a8f36de82e1278abb02f",
    "response_code": 1,
    "scan_date": "2014-04-14 21:17:17",
    "permalink":
"https://www.virustotal.com/file/275a021b
bfb6489e54d471899f7db9d1663fc695ec2
fe2a2c4538aabf651fd0f/analysis/139751
0237/",
    "verbose_msg": "Scan finished, scan
information embedded in this object",
    "sha256":
"275a021bbfb6489e54d471899f7db9d16
63fc695ec2fe2a2c4538aabf651fd0f",
    "positives": 49,
    "total": 51,
    "md5":
"44d88612fea8a8f36de82e1278abb02f",
    "scans": {
        "Bkav": {
            "detected": true,
            "version": "1.3.0.4959",
            "result": "DOS.EiracA.Trojan",
            "update": "20140412"
        },
        "MicroWorld-eScan": {
            "detected": true,
            "version": "12.0.250.0",
            "result": "EICAR-Test-File",
            "update": "20140414"
        },
        "nProtect": {
            "detected": true,
            "version": "2014-04-14.02",
            "result": "EICAR-Test-File",
            "update": "20140414"
        },
        ...<snip>...
        "AVG": {
            "detected": true,
            "version": "13.0.0.3169",
            "result": "EICAR_Test",
            "update": "20140414"
        },
        "Panda": {
            "detected": true,
            "version": "10.0.3.5",
            "result":
"EICAR-AV-TEST-FILE",
            "update": "20140414"
        },
        "Qihoo-360": {
            "detected": true,
            "version": "1.0.0.1015",
            "result": "Trojan.Generic",
            "update": "20140414"
        }
    }
}
```


```
$ ./tests
```