Centro de
Investigación en
Computación

Laboratory of Malware

Laboratorio de malware

Sistemas

Técnicas

Etc.

Course

Análisis y Detección de Malware

Instructor

Acosta Bermejo Raúl

Lecture notes

2023-A
7 de noviembre del 2024

Instituto
Politécnico
Nacional

Table of contents (outline)

Tabla de contenido

1. Introducción
2. Hipervisores
3. Contenedores
4. Casos de Estudio
    1. Cuckoo

Introducción

Bibliografía

Links

- Teoria
  - https://www.sciencedirect.com/topics/computer-science/malware-laboratory
- Ejemplos en Universidades
  - Israel. https://www.malware-lab.com/.
  - Republic of Azerbaijan. https://cert.gov.az/en/malware-lab
- Blogs
  - https://cybersecurity.att.com/blogs/security-essentials/building-a-home-lab-to-become-a-malware-hunter-a-beginners-guide
  - https://zeltser.com/build-malware-analysis-toolkit/
- Papers
  - Creating a Malware Analysis Lab and Basic Malware Analysis. Joseph Peppers, 2018 Iowa State University

Introduction

Banco de pruebas

Testbed (definición)

The term is used across many disciplines to describe experimental research and new product development platforms and environments.

1. A piece of equipment used for testing new machinery, especially aircraft engines.
2. It is a **platform** for conducting rigorous, transparent, and replicable testing of scientific theories, computational tools, and new technologies.

Entorno controlado

Conceptos

Herramientas

Entorno controlado

Conceptos

- Sensores / Agentes de monitoreo
  - Captura de datos generados: paquetes de red, archivos modificados, registros o bases de datos (*Windows Registry*), etc.
- Limitar el acceso a recursos:
  - Sistema de archivos
  - Dispositivos de red y/o lo que se transmite
- Control de inicio, suspensión, y fin.
  - Guardar estados de ejecución: reinicios.
- Controles externos
  - Subredes y reglas de enrutamiento
- Copias de la memoria
  - De todo (snaphots) o solo la memoria (Forense con Volatility).

Análisis Estático

Análisis Dinámico

Entorno controlado

Antecedentes

- Jaulas / Jail

  Su origen viene de FreeBSD.

También conocido como
Sandboxing

- **Definición**

  Es una operación que **cambia el directorio raiz** para el proceso actual en ejecución y sus hijos. El programa que corre en este ambiente modificado no puede acceder los archivos que están fuera del directorio designado. La idea es crear un directorio con todo lo necesario (copia o link) para que el proceso se ejecute. Después se usa el syscall chroot para cambiar el directorio raiz. Como no se pueden referenciar trayectorias fuera del nuevo raiz, esto impide leer o escribir archivos de manera maliciosa.

- Ejemplos
  - Chroot (like-UNIX)
  - Sandboxie (Windows)

Entorno controlado

Máquinas Virtuales

Intel® 64 and IA-32 Architectures
Software Developer's Manual

Volume 1:
Basic Architecture

- Hipervisor
  - Motor de ejecución: inicio, suspensión, parar, clonar, destruir.
  - Control local o remoto: GUI, CLI, API.
- Implementaciones
  - Software: Wine.
  - Hardware: Instrucciones del procesador

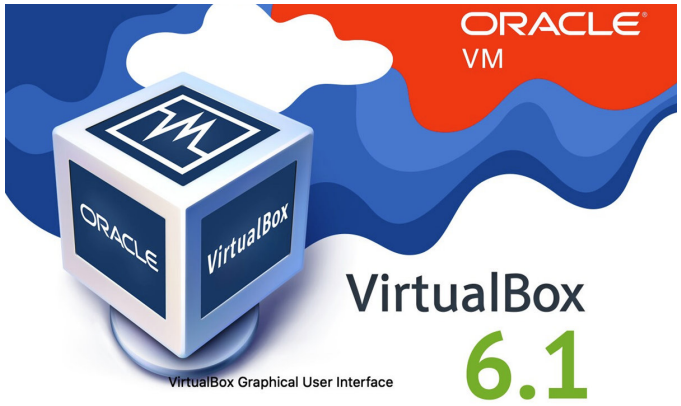
    AMD-v, Intel – VT-x,i,d
- Lecturas
  - *White paper*: Enabling Intel® Virtualization Technology Features and Benefits.
  - Manuales: https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html

Entorno controlado

Máquinas Virtuales

https://www.vmware.com/

Ejemplos de hipervisores



https://www.virtualbox.org/

Microsoft Hyper-v

https://docs.microsoft.com/en-us/virtualization/

https://www.qemu.org/

https://xenserver.org/

KVM (XenServer, OpenNebula).

Entorno controlado

Contenedores

- Definición

  Es un espacio de usuario aislado en el cual se ejecuta un programa.

  El SO controla el acceso (restringe) los recursos disponibles: archivos, dispositivos de E/S, etc.

- Ejemplos
  - Docker. Multiplataforma
  - LXC/LXD. Sólo en Linux.
- Usos del sanboxing
  - Para desarrollar nuevas aplicaciones: con baterias pruebas.
  - Para analizar malware, por ejemplo:

    https://cwsandbox.org/

Hypervisors

Hipervisores y Máquinas Virtuales

Hypervisor

Hipervisores

Definition

A hypervisor, also known as a virtual machine monitor or VMM, is software that creates and runs virtual machines (VMs). A hypervisor allows one host computer to support multiple guest VMs by virtually sharing its resources, such as memory and processing.

Types

- Type 1 or bare metal.
  It acts like a lightweight operating system and runs directly on the host's hardware,
- Type 2 or hosted.
  It runs as a software layer on an operating system, like other computer programs.

Links

- https://www.vmware.com/topics/glossary/content/hypervisor
- https://en.wikipedia.org/wiki/Hypervisor
- https://www.redhat.com/en/topics/virtualization/what-is-a-hypervisor

Hypervisor

Administrador de hipervisores

Vagrant

- It is an open-source software product for building and maintaining portable virtual development environments.
- The core idea behind its creation lies in the fact that the environment maintenance becomes increasingly difficult in a large **project** with multiple technical stacks.
- Vagrant manages all the necessary configurations for the developers in order to avoid the unnecessary maintenance and setup time, and increases development productivity.
- Vagrant is written in the Ruby language but its ecosystem supports development in almost all major languages.
- Vagrant uses "Provisioners" and "Providers" as building blocks to manage the development environments. Some of the most widely used Providers are VirtualBox, Amazon AWS, VMWare, and Docker.

Link

- https://www.vagrantup.com/

Containers

Contenedores

Containers

Open Source

Some of the main containers are:

- **Docker** is an open-source project that automates the deployment of Linux applications inside software containers.
- **LXC** is an operating-system-level virtualization method for running multiple isolated Linux systems (containers) on a control host using a single Linux kernel.

Containers

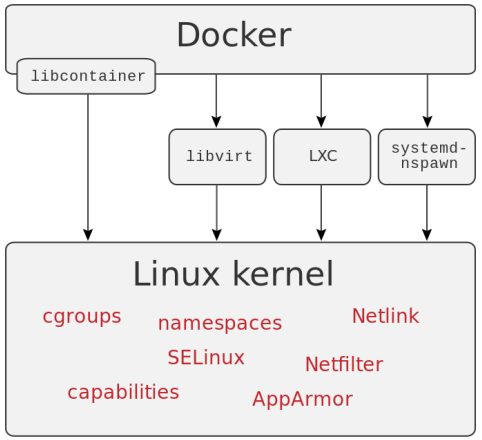
Open Source

Docker

- Docker provides an additional layer of abstraction and automation of operating-system-level virtualizationon Linux.
- Docker uses the resource isolation features of the Linux kernel such as cgroups and kernel namespaces, and a union-capable file system such as aufs and others to allow independent "containers" to run within a single Linux instance, avoiding the overhead of starting and maintaining virtual machines.

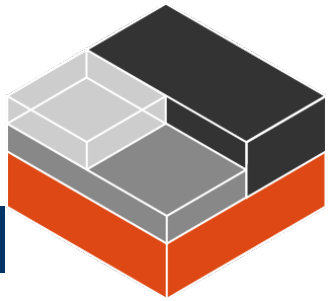
Link

- https://en.wikipedia.org/wiki/Docker_(software)
- https://www.docker.com/



Containers

Open Source



LXC

- It is an operating-system-level virtualization method for running multiple isolated Linux systems (containers) on a control host using a single Linux kernel.
- The Linux kernel provides:
  - The cgroups functionality that allows limitation and prioritization of resources (CPU, memory, block I/O, network, etc.) without the need for starting any virtual machines, and
  - The namespace isolation functionality that allows complete isolation of an applications' view of the operating environment, including process trees, networking, user IDs and mounted file systems.
- LXC combines kernel's cgroups and support for isolated namespaces to provide an isolated environment for applications.

Link

- https://linuxcontainers.org/

Study Cases

Casos de Estudio

5 casos

Labs

Casos de Estudio

Some of the main are:

1. On the cloud
    1. VirusTotal
    2. hybrid-análisis
    3. Etc.
2. Local
    i. Cuckoo Sandbox.
    ii. MultiScanner
    iii. REMnux
    iv. Angr

Cloud Services

Servicios en la Nube

No requiere ninguna instalación

VirusTotal

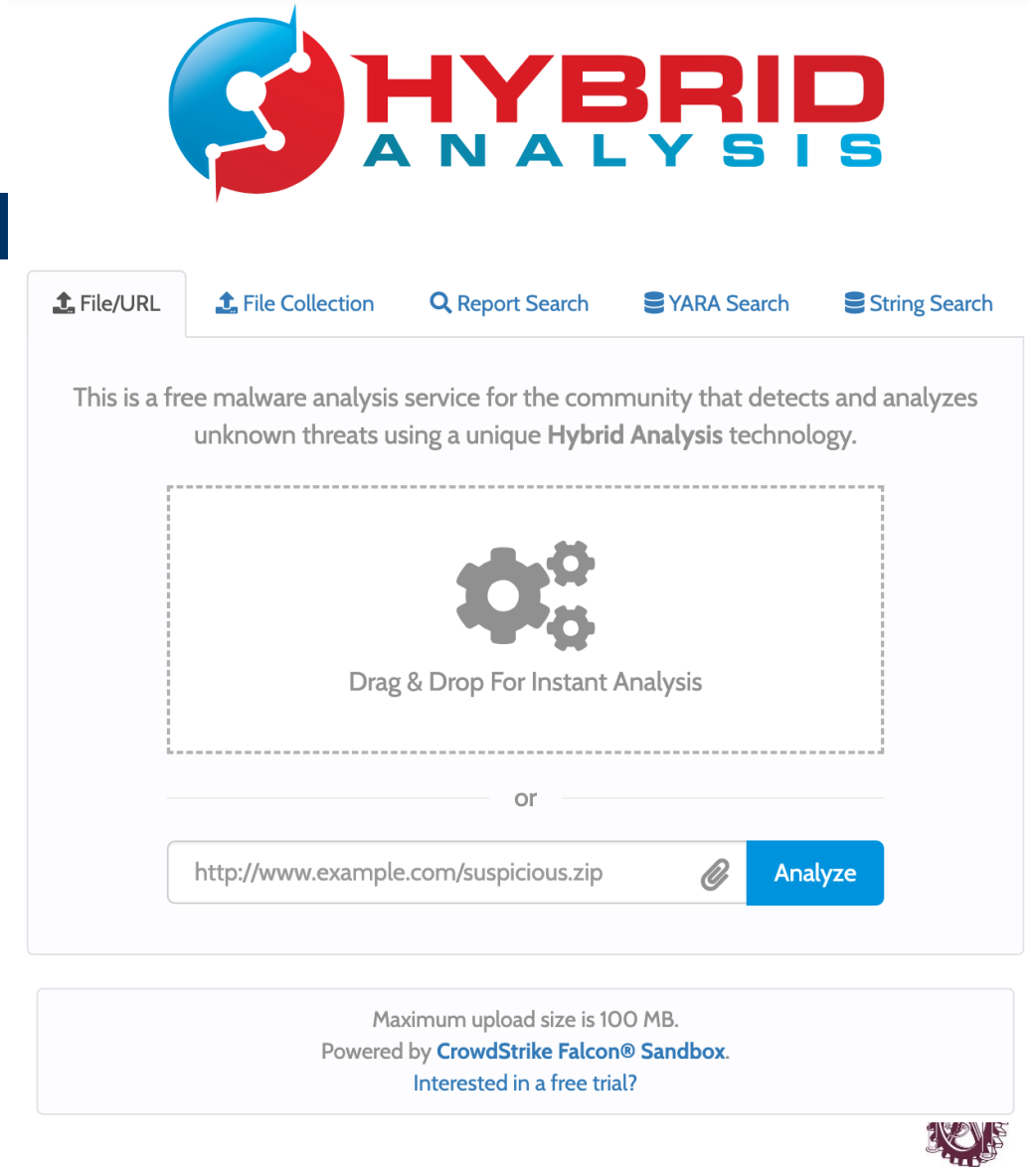
Ejemplo 1

- Referencias
  - https://www.virustotal.com/

Hybrid

Ejemplo 2

- Referencias
  - https://www.hybrid-analysis.com/.



ANY.run

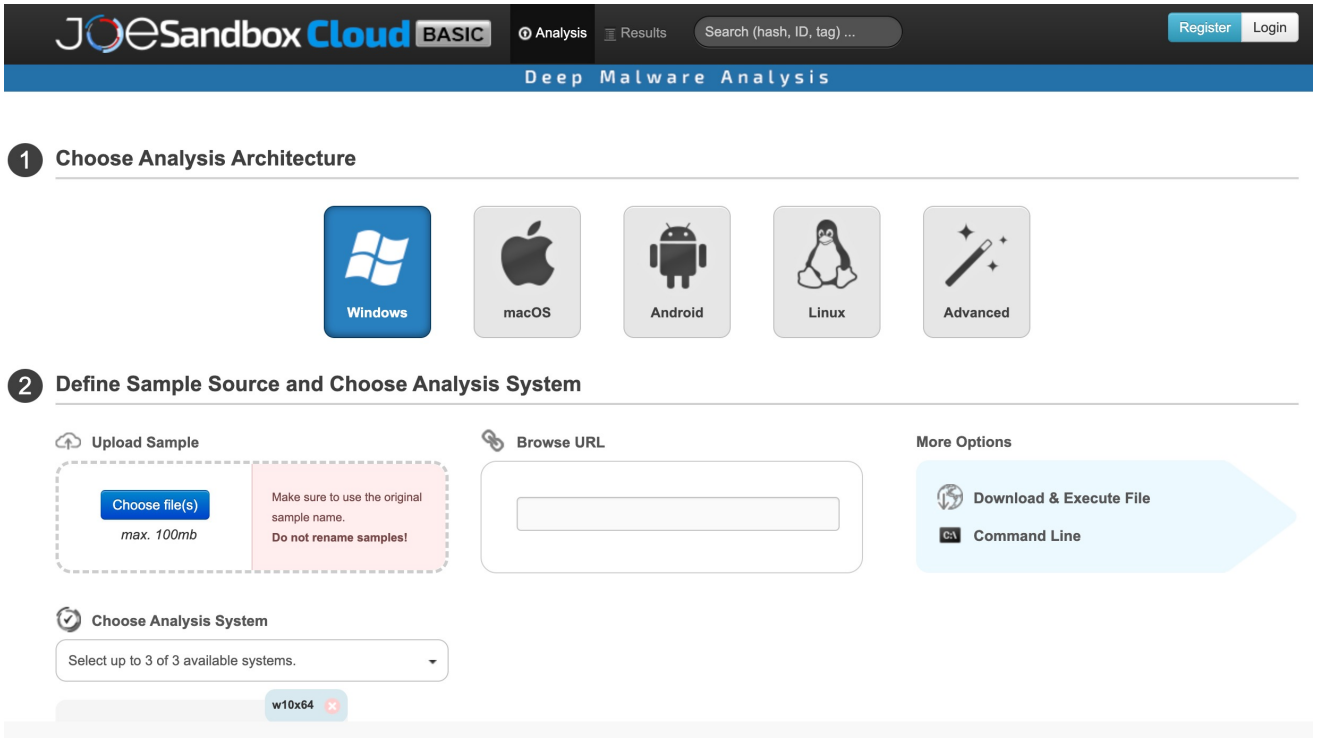
Ejemplo 3

- Referencias
  - https://any.run/

Joe Sandbox

Ejemplo 4

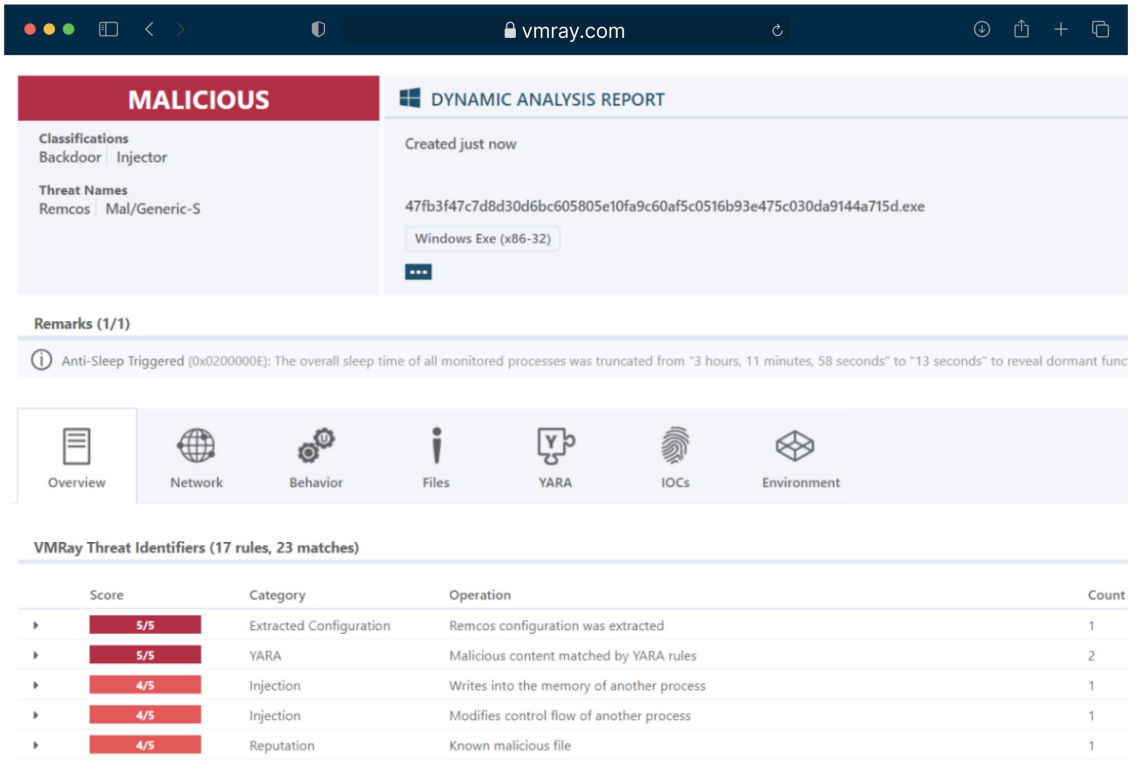
- Referencias
  - https://www.joesandbox.com/#windows



VMray

Ejemplo 5

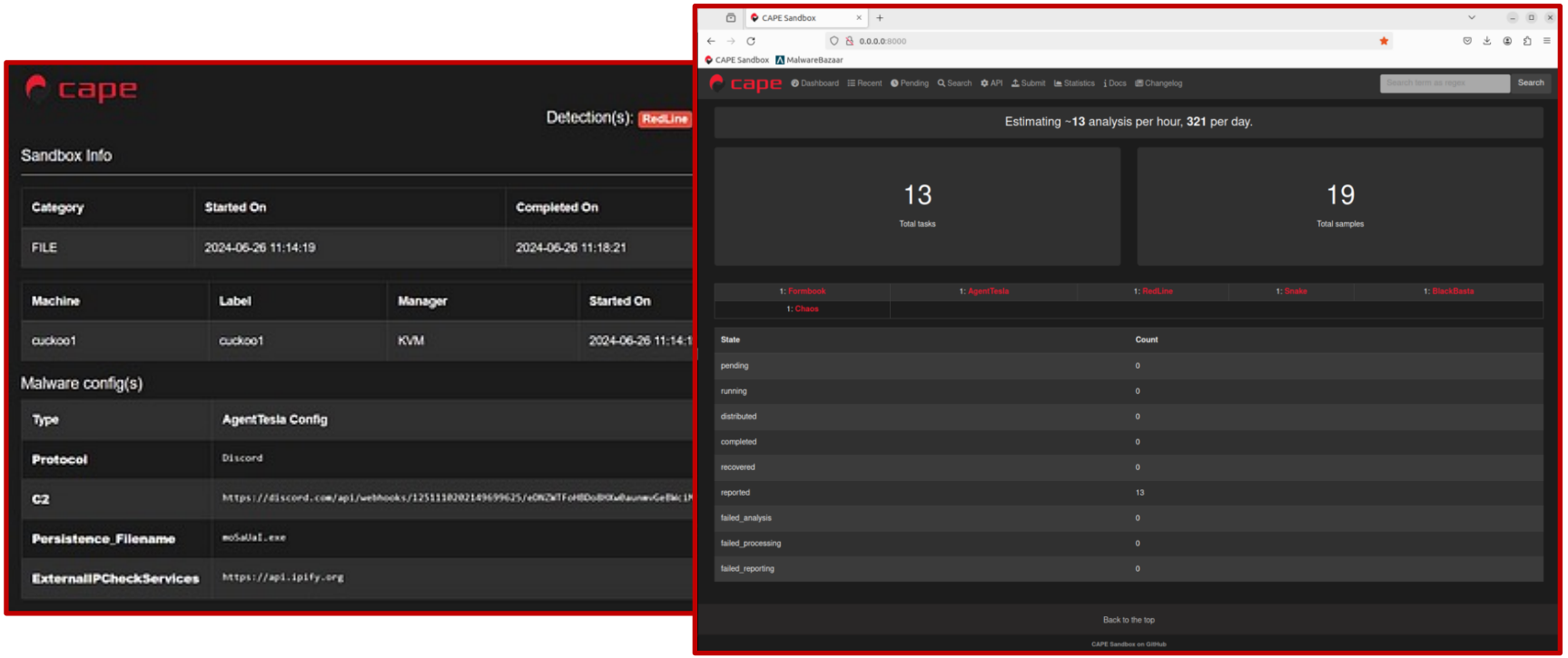
- Referencias
  - https://www.vmray.com/



CAPE Sandbox

Ejemplo 6

- Referencias
  - https://capev2.readthedocs.io/en/latest/
  - https://endsec.au/blog/building-an-automated-malware-sandbox-using-cape/



Cuckoo

Filosofía de diseño

Arquitectura

Programación, API

Cuckoo

Introducción

Cuckoo

Arquitectura

- Virtual Machine
  - Puede usar varios Hipervisores: VirtualBox, Vmware, Qemu.
- Scheduler
  - Es el componente principal.
  - Programado en Python (100%), y fácilmente “customizable”.
  - Ejecuta tareas de un pool de máquinas virtuales disponibles.
  - Las tareas se ejecutan con base en los módulos que integran varias herramientas: análisis estático y dinámico.
- Analyzer
  - Son componentes (Cmonitor, Chook) programados con python para procesar la información capturada.

Cuckoo

Arquitectura

URLs

- Versiones en la nube:
  - https://malwr.ee/
- .

Cuckoo

Herramientas

Lista (No exhaustiva)

- Análisis de archivos ejecutables
  - Ejecutables y librerías, Documentos de Ms Office, Java (Jar), PDF, Comprimidos (Zip y otros empaquetadores).
- Captura de datos de red
  - Tcpdump, Wireshark (Analizador de Red, Modo promiscuo)
- Captura de la memoria RAM
  - Volatility

Ejecución remota (SSH) y automatizada de las herramientas mediante python.

Cuckoo

Captura de la memoria RAM

Volatility 2.6 (Windows 10 / Server 2016)

This release improves support for Windows 10 and adds support for Windows Server 2016, Mac OS Sierra 10.12, and Linux with KASLR kernels. A lot of bug fixes went into this release as well as performance enhancements (especially related to page table parsing and virtual address space scanning). See below for a more detailed list of the changes in this version.

This release also coincides with the Community repo - a collection of Volatility plugins written and maintained by authors in the forensics community. Many of these are the result of the last 4 years of Volatility plugin contests, but some were just written for fun. Either way, its an entire arsenal of plugins that you can easily extend into your existing Volatility installation.

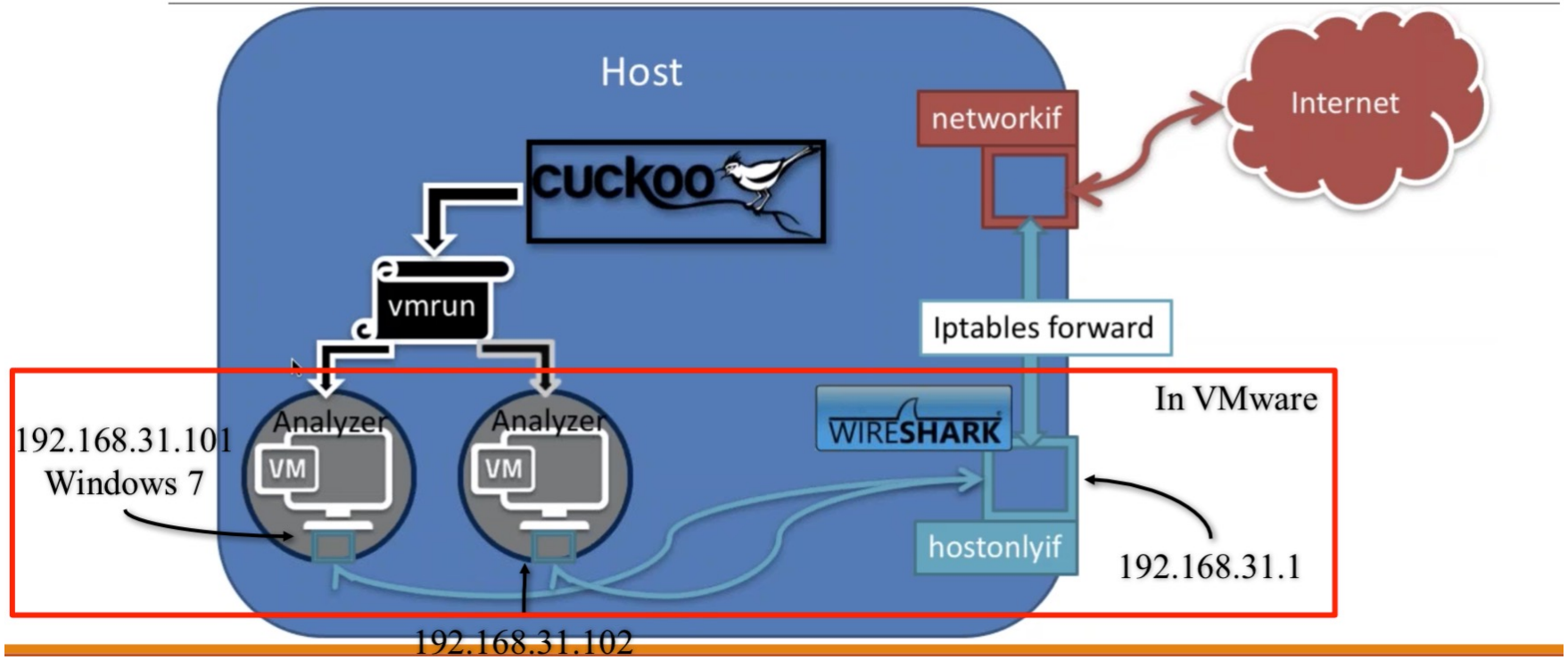
Released: December 2016

- Volatility 2.6 Windows Standalone Executable (x64)
- Volatility 2.6 Mac OS X Standalone Executables (x64)
- Volatility 2.6 Linux Standalone Executables (x64)
- Volatility 2.6 Source Code (.zip)
- Integrity Hashes
- View the README
- View the CREDITS

Se utiliza mediante rutinas de python

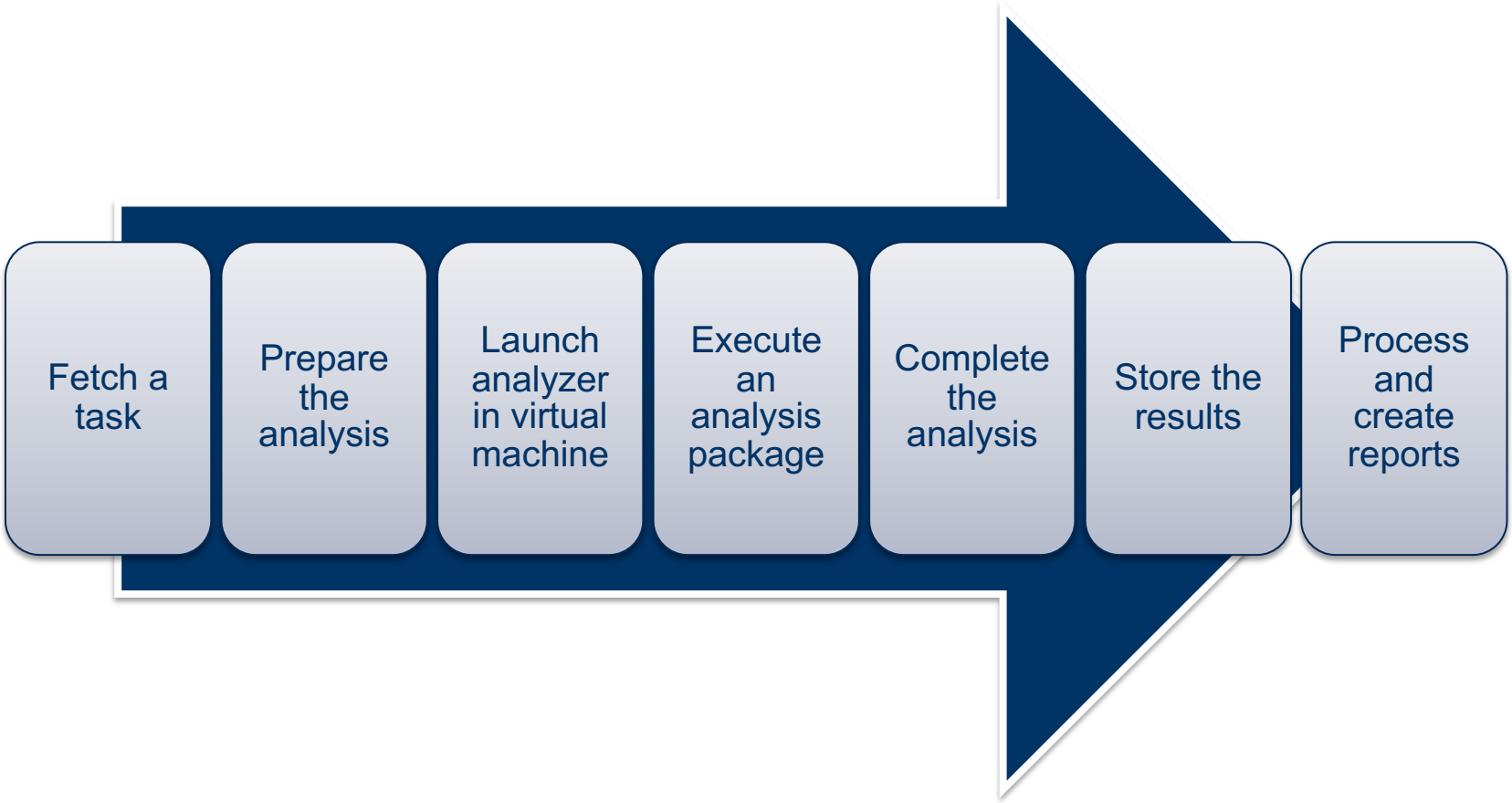
Cuckoo

Arquitectura



Cuckoo

Flujo de ejecución



Cuckoo

Programación con Python

```python
from cuckoo.common.abstracts import Machinery
from cuckoo.common.exceptions import CuckooMachineError

class MyMachinery(Machinery):
    def start(self, label):
        try:
            revert(label)
            start(label)
        except SomethingBadHappens:
            raise CuckooMachineError("oops!")

    def stop(self, label):
        try:
            stop(label)
        except SomethingBadHappens:
            raise CuckooMachineError("oops!")
```

The only requirements for Cuckoo are that:

- The class inherits from `Machinery`.
- You have a `start()` and `stop()` functions.
- You raise `CuckooMachineError` when something fails.

Cuckoo

API REST

- Utiliza un servidor de páginas web: nginx
- Usa por default la dirección:
  localhost:8000

Resource	Description
POST /tasks/create/file	Adds a file to the list of pending tasks to be processed and analyzed.
POST /tasks/create/url	Adds an URL to the list of pending tasks to be processed and analyzed.
POST /tasks/create/submit	Adds one or more files and/or files embedded in archives to the list of penc
GET /tasks/list	Returns the list of tasks stored in the internal Cuckoo database. You can op
GET /tasks/sample	Returns the list of tasks stored in the internal Cuckoo database for a given s
GET /tasks/view	Returns the details on the task assigned to the specified ID.
GET /tasks/reschedule	Reschedule a task assigned to the specified ID.
GET /tasks/delete	Removes the given task from the database and deletes the results.
GET /tasks/report	Returns the report generated out of the analysis of the task associated with
GET /tasks/screenshots	Retrieves one or all screenshots associated with a given analysis task ID.
GET /tasks/rereport	Re-run reporting for task associated with a given analysis task ID.
GET /tasks/reboot	Reboot a given analysis task ID.
GET /memory/list	Returns a list of memory dump files associated with a given analysis task ID
GET /memory/get	Retrieves one memory dump file associated with a given analysis task ID.
GET /files/view	Search the analyzed binaries by MD5 hash, SHA256 hash or internal ID (ref
GET /files/get	Returns the content of the binary with the specified SHA256 hash.
GET /pcap/get	Returns the content of the PCAP associated with the given task.

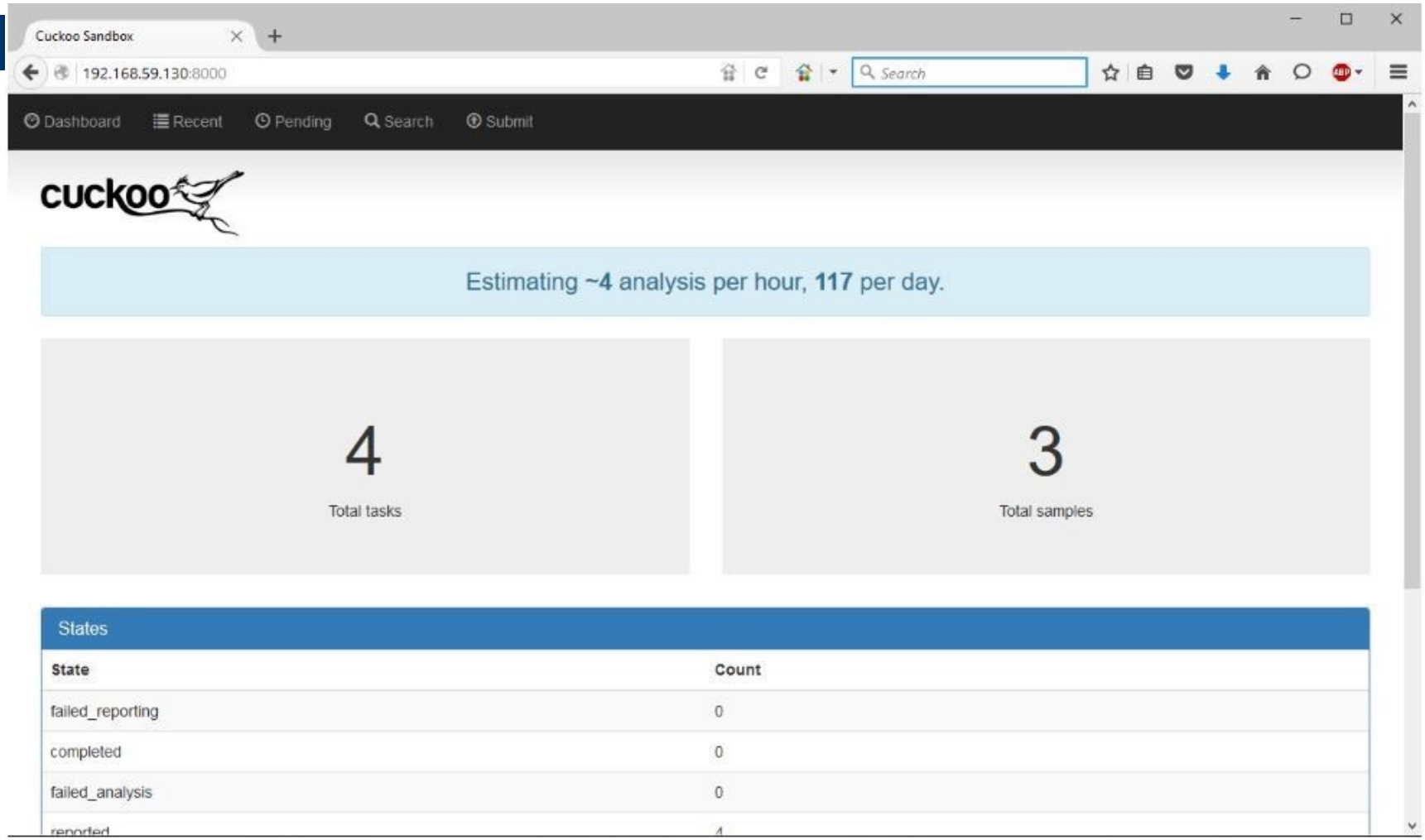
Cuckoo

Web


File Edit View History Bookmarks Tools Help
Cuckoo Sandbox
localhost:8080
Google
cuckoo
Home
Search for MD5:
Search
Recent Analysis
Added On
MD5
Target
Analysis Package
2011-12-18 17:20:15
9b2de8b062a5538d2a126ba93895d1e9
/tmp/malware.exe
None
©2010-2011 Cuckoo Sandbox.

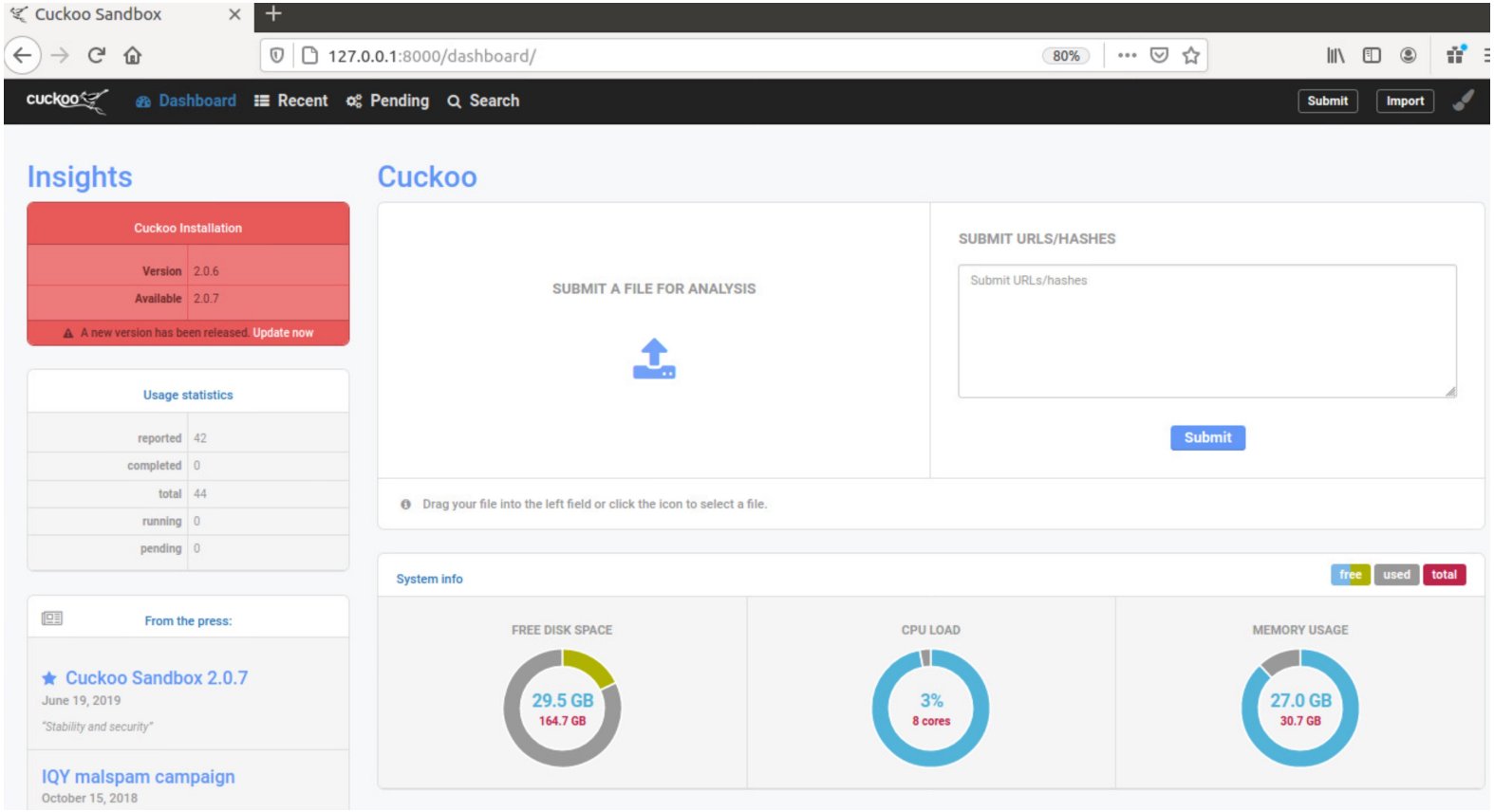
Cuckoo

Web


Cuckoo Sandbox
192.168.59.130:8000
Search
Dashboard
Recent
Pending
Search
Submit
cuckoo
Estimating ~4 analysis per hour, 117 per day.
4
Total tasks
3
Total samples
States
State
Count
failed_reporting
0
completed
0
failed_analysis
0

Cuckoo

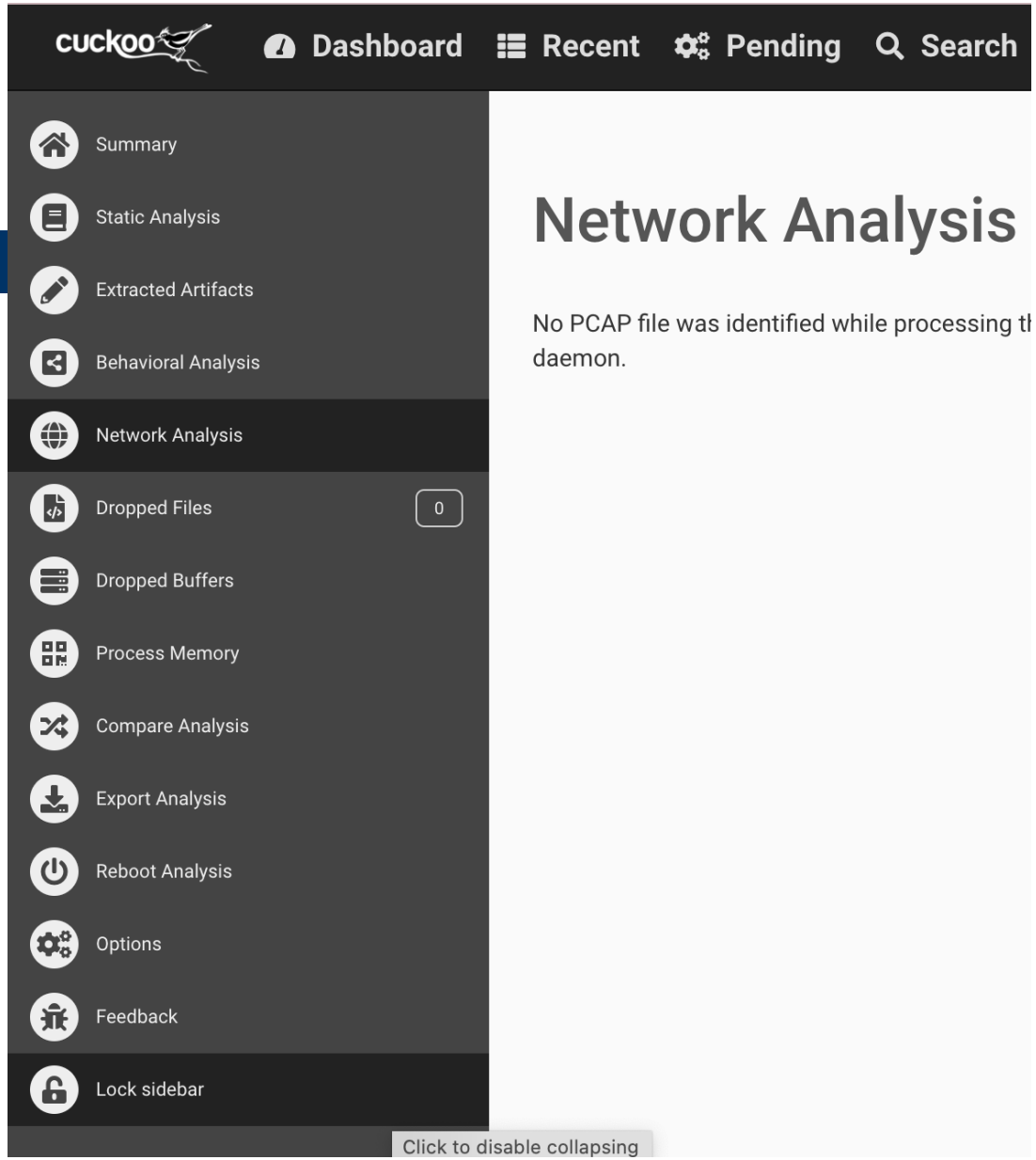
Web


Cuckoo Sandbox
127.0.0.1:8000/dashboard/
80%
cuckoo
Dashboard
Recent
Pending
Search
Submit
Import
Insights
Cuckoo Installation
Version 2.0.6
Available 2.0.7
A new version has been released. Update now
Usage statistics
reported 42
completed 0
total 44
running 0
pending 0
From the press:
Cuckoo Sandbox 2.0.7
June 19, 2019
"Stability and security"
IQY malspam campaign
October 15, 2018
Cuckoo
SUBMIT A FILE FOR ANALYSIS
SUBMIT URLS/HASHES
Submit URLs/hashes
Submit
Drag your file into the left field or click the icon to select a file.
System info
free
used
total
FREE DISK SPACE
29.5 GB
164.7 GB
CPU LOAD
3%
8 cores
MEMORY USAGE
27.0 GB
30.7 GB

Cuckoo

Web

Después de que se sube un archivo y se termina de analizar, cuckoo da los resultados y al verlos ofrece un menú nuevo de resultados.



Cuckoo

Web

Interacción con el servidor

Terminal
$ cuckoo submit $path/file

API REST
$ wget --header='Authorization: 12345678’ http://localhost:8090/machines/list

Petición HTTP enviada, esperando respuesta... 401 UNAUTHORIZED
La autentificación usuario/contraseña falló.

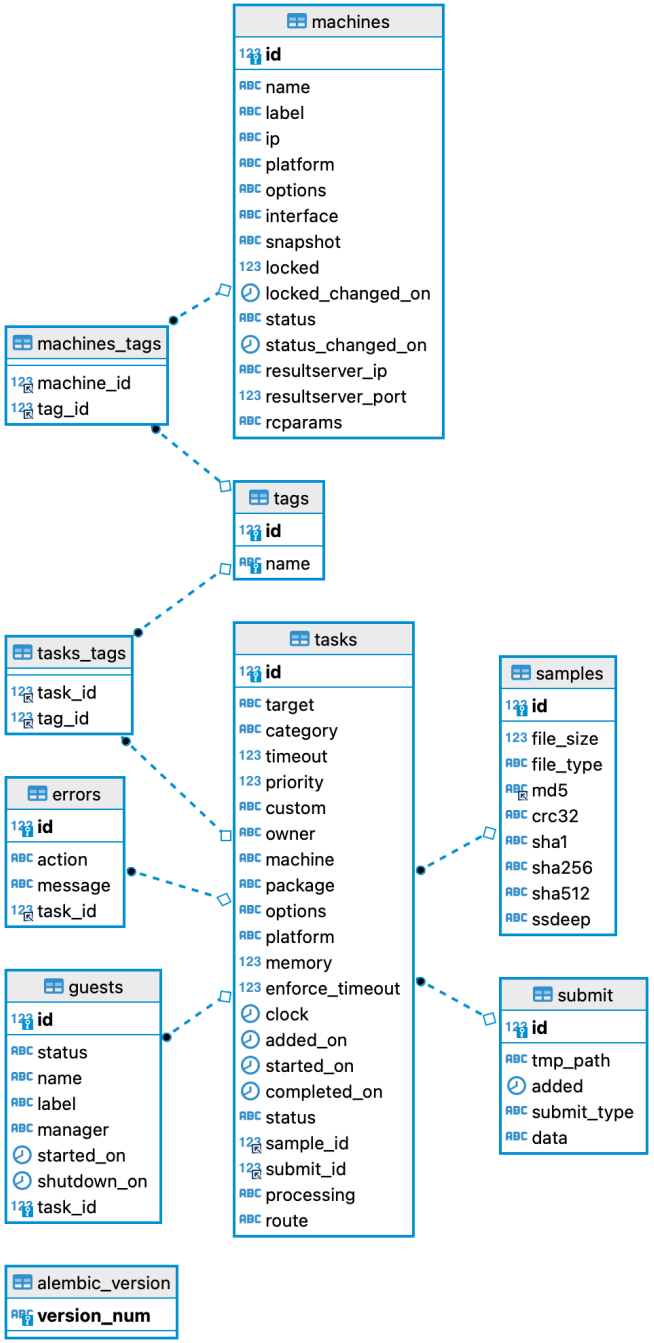
$ curl -H 'Authorization: 12345678' http://localhost:8090/machines/list

{"message":"Authentication in the form of an 'Authorization: Bearer <TOKEN>' header is required"}

Cuckoo

BD

Diagrama ER



Cuckoo

Referencias

- Sitios oficiales
  - https://cuckoosandbox.org/
  - https://github.com/cuckoosandbox/cuckoo
- Documentación y artículos
  - https://cuckoo.readthedocs.io/en/latest/introduction/what/
  - IEEE. Matching and retrieval of state machine diagrams from software repositories using Cuckoo Search Algorithm
    https://ieeexplore.ieee.org/document/8079998
- Videos y cursos
  - https://www.youtube.com/watch?v=V4z2tLRCuIY
- Ejemplos de instalación
  - https://cuckoo.cert.ee/

Cuckoo

Instalación

Instalación

- Versión: 2.0.7
- Links
  - https://cuckoo.sh/docs/installation/index.html
  - Host y Guest
    - https://cuckoo.sh/docs/installation/host/index.html
    - https://cuckoo.sh/docs/installation/guest/index.html

Cuckoo

Instalación y Ejecución

- Instalar en un ambiente (env) de python
  - $ cd my-project-cuckoo/
  - $ virtualenv venv
  - $ source venv/bin/activate
  - (venv) $
  - $ pip install <package> $ pip install -U cuckoo
  - ..
  - $ deactivate
- Cuckoo Working Directory (CWD).
  - All configurable components, generated data, and results of Cuckoo will be stored in this directory.
  - The first time you run Cuckoo a CWD checkout will be created for you automatically:
  - $ **cuckoo –d**
  - $ ls -la $HOME/.cuckoo $HOME/.cuckoo/conf

CWD defaults to ~/.cuckoo

Ejemplos de archivos de configuración:

cuckoo.conf
auxiliary.conf
virtualbox.conf
processing.conf
reporting.conf

El módulo de reportes se puede desactivar y Guarda la info en la BD

Cuckoo

Instalación y Ejecución

CRITICAL: CuckooCriticalError: Unable to bind ResultServer on 192.168.56.1:2042 [Errno 49] Can't assign requested address. This usually happens when you start Cuckoo without bringing up the virtual interface associated with the ResultServer IP address. Please refer to https://cuckoo.sh/docs/faq/#troubles-problem for more information.

Se usa para guardar los reportes
No los datos de los análisis

- Servidor Web
  - $ **cuckoo web runserver**
  - En el navegador cargar la página: localhost:8000 o 127.0.0.1:8000
    - In order to use the Cuckoo Web Interface it is required to have **MongoDB** up-and-running and enabled in Cuckoo. Please refer to our official documentation as well as the $CWD/conf/reporting.conf file
  - Dependienod del SO hacer la instalación de MongoDB Community Server
    Hay la versión Cloud (Mongo DB Atlas) pero es de paga o limitada.
    Verificar las **bitácoras** (logs): tree ~/.mongodb/
- Result Server
  - Se usa para recibir en tiempo real los resultados (logs) del analyzer.
  - Usa la IP definida en cuckoo que por default es 192.168.56.1 y el puerto 2042.
    La información esta en cuckoo.conf
  - Para que sea accesible hay que:
    - Cambiarla según la red local (lo que tenga el Access Point).
    - Desactivar el Firewall si es necesario.

Cuckoo

Instalación y Ejecución

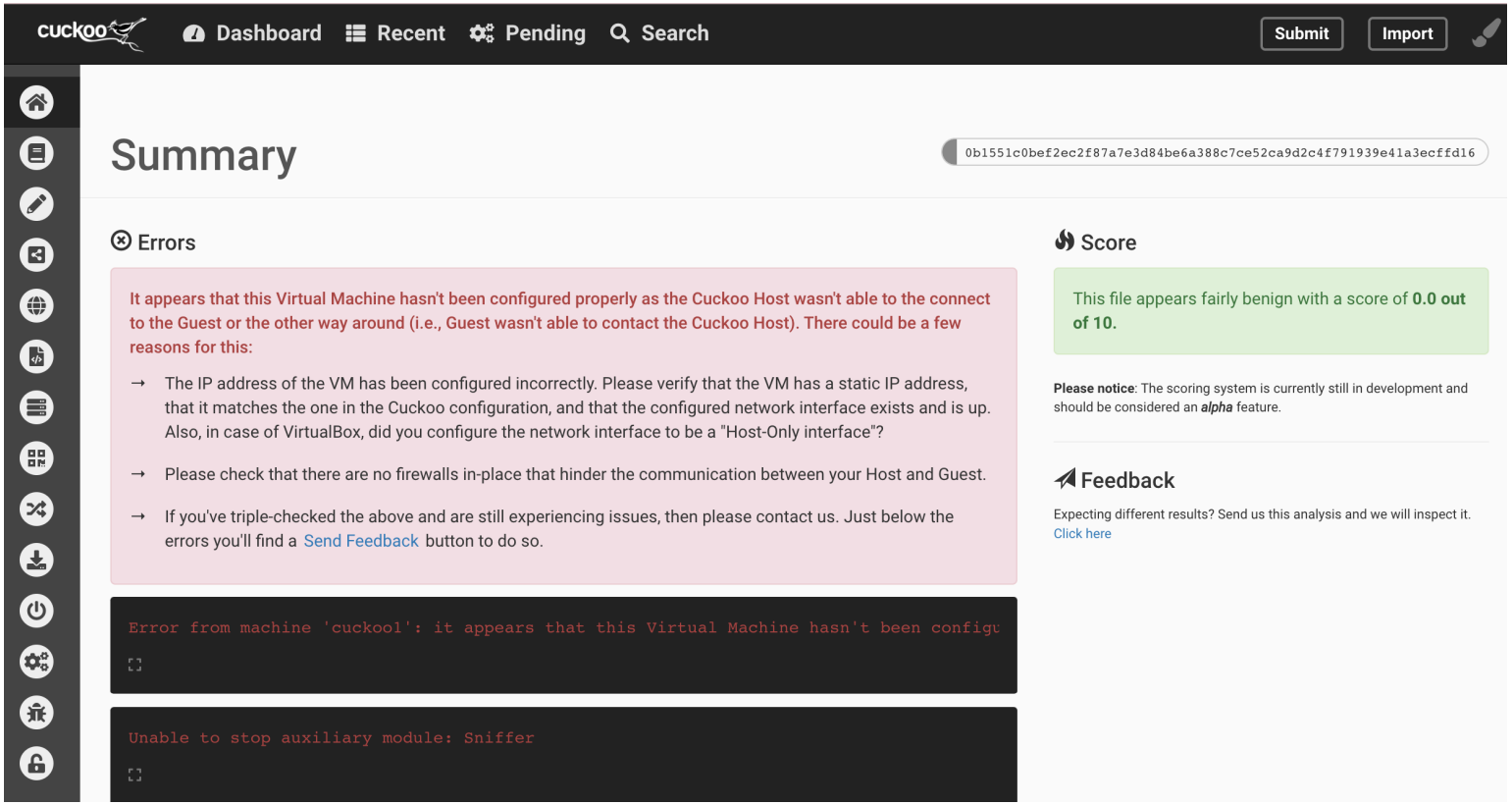
- Base de Datos
  - No es la de MongoDB.
  - **cuckoo.conf**: connection = mysql://usuario:password@localhost/database
  - [cuckoo] CRITICAL: CuckooDependencyError: Missing MySQL database driver (install with `pip install mysql-python`
- Si se necesitan varias instalaciones
  - $ sudo mkdir /opt/cuckoo
  - $ sudo chown cuckoo:cuckoo /opt/cuckoo
  - $ cuckoo --cwd /opt/cuckoo

    # You could place this line in your .bashrc, for example.
  - $ export CUCKOO=/opt/cuckoo
  - $ cuckoo

Cuckoo

Instalación y Ejecución

- A pesar de configurar la MV



Cuckoo

Instalación y Ejecución

- Problemas típicos
  - Cambian la configuración (por ejemplo de MongoDB) y no reiniciar cuckoo.
- Otras referencias
  - VMCloack
    - It is a utility for automatically creating Virtual Machines with Windows as guest Operating System.

    https://hatching.io/blog/cuckoo-sandbox-setup/

    https://vmcloak.readthedocs.io/en/latest/
  - Blog
    - Describe el proceso en Windows
    - https://medium.com/@oshara.16/setting-up-cuckoo-sandbox-for-dummies-malware-analysis-3daa99e950b5.
  - https://arnaudloos.com/2019/cuckoo-sandbox-installation/

Cuckoo

Instalación y Ejecución

Resumen

- Instalación
  - Ambiente python
    - Versión 2.7 con los módulos: mysql, crypto, etc.
  - VirtualBox
    - Crear una MV con el nombre adecuado.
    - Instalar el GuesAddtions.
    - Crear un snapshot.
    - Configurar la dirección IP para el Servidor de Resultados
  - MongoDB: para los reportes
  - MySQL: para los datos de los análisis.

Cuckoo

Temas avanzados

Como se realiza el análisis?

- Zer0m0n
  - It is a driver for Cuckoo Sandbox, it will perform kernel analysis during the execution of a malware. There are many ways for a malware author to bypass Cuckoo detection, he can detect the hooks, hardcodes the Nt* functions to avoid the hooks, detect the virtual machine... The goal of this driver is to offer the possibility for the user to choose between the classical userland analysis or a kernel analysis, which will be harder to detect or bypass.
- https://github.com/angelkillah/zer0m0n

- Leer el blog de los desarrolladores
  - Ofrecen servicios de seguridad.
  - Platican sobre los cambios y problemas de cuckoo.
- https://cuckoo.sh/blog/

MultiScanner

Framework

Listado

MultiScanner

Introducción

- Desarrollado por el MITRE (empresa de EU sin lucro).
  - Desarrollo el CVE
- Características
  - Workflow. Genera Json
  - Integra varias herramientas
  - Cola de tareas (nodos con Celery/RabbitMQ)
  - Web UI, REST API, Reportes.
  - Interactúa con ElastiSearch
- Links
  - https://multiscanner.readthedocs.io
  - https://github.com/mitre/multiscanner

MultiScanner

Introducción

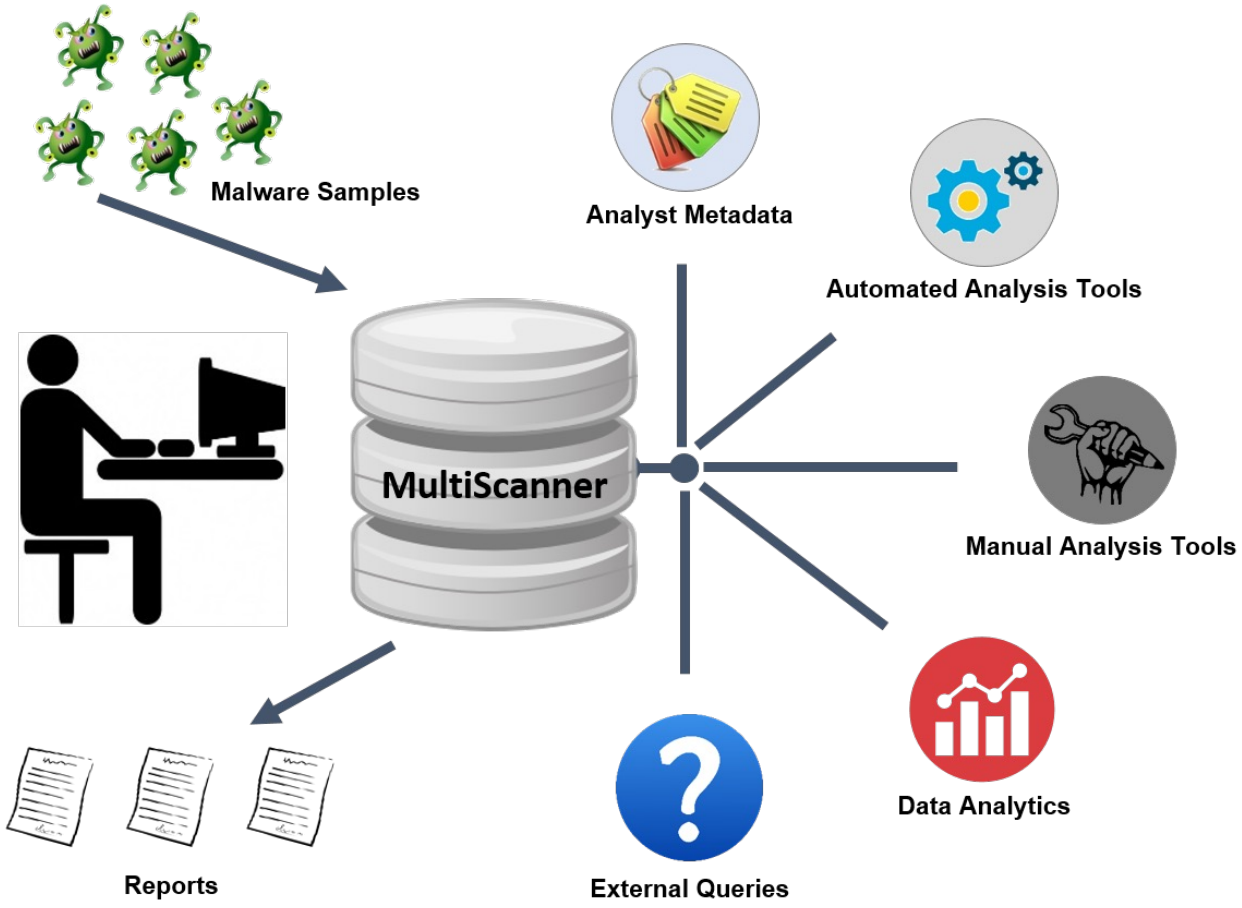
Ejemplos de herramientas

- Identificador de archivos
  - Comandos File, TrID
  - http://mark0.net/soft-trid-e.html
- VirusTotal
  - Script vtsearch: searches VirusTotal for sample's hash and downloads the report if available.
- Antivirus
  - ClamAV, YARA, etc.
- Otras plataformas de análisis
  - Cuckoo, VxStream, FireEye.
- Analizador de ejecutables
  - PEfile.

MultiScanner

Introducción

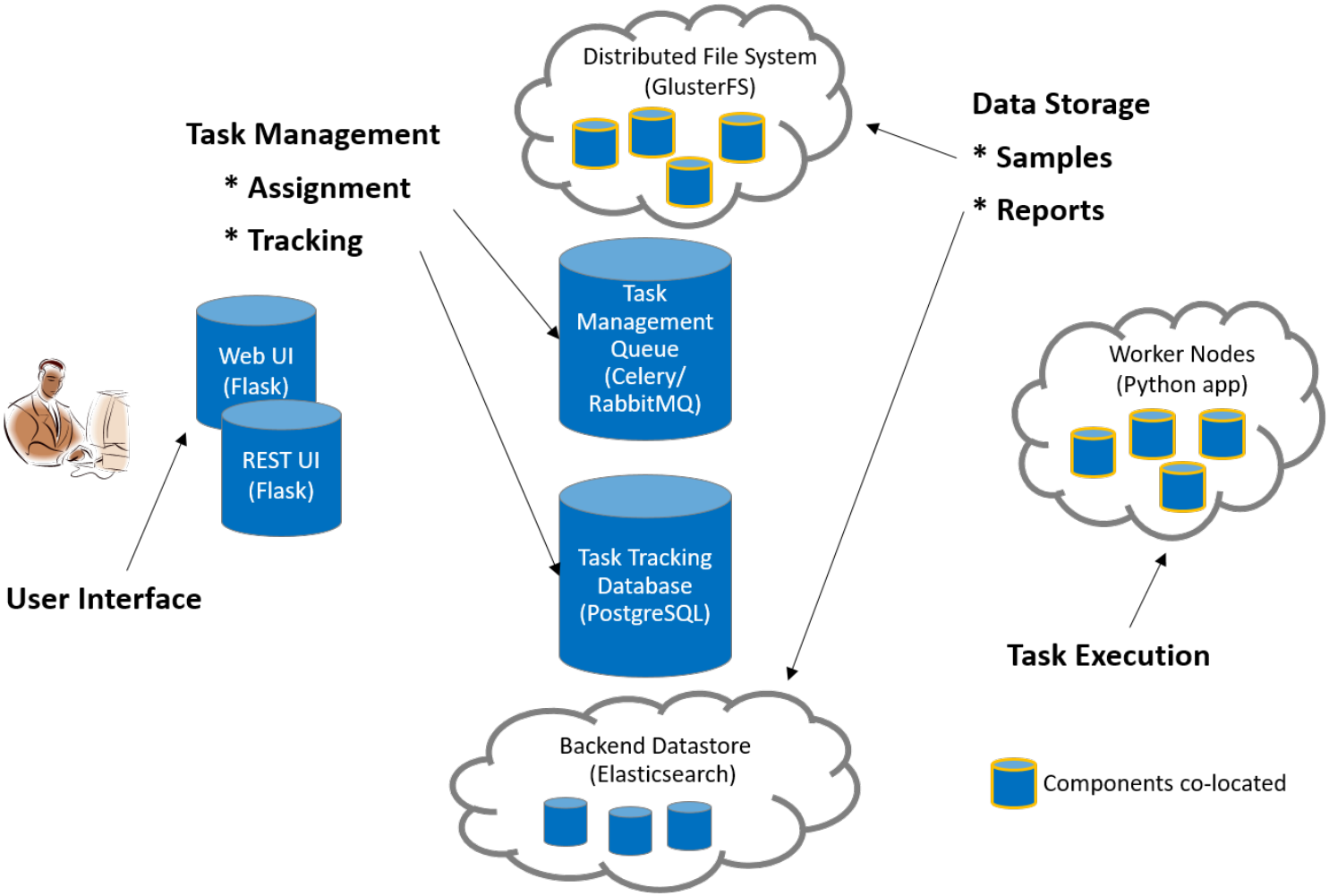
Arquitectura



MultiScanner

Introducción

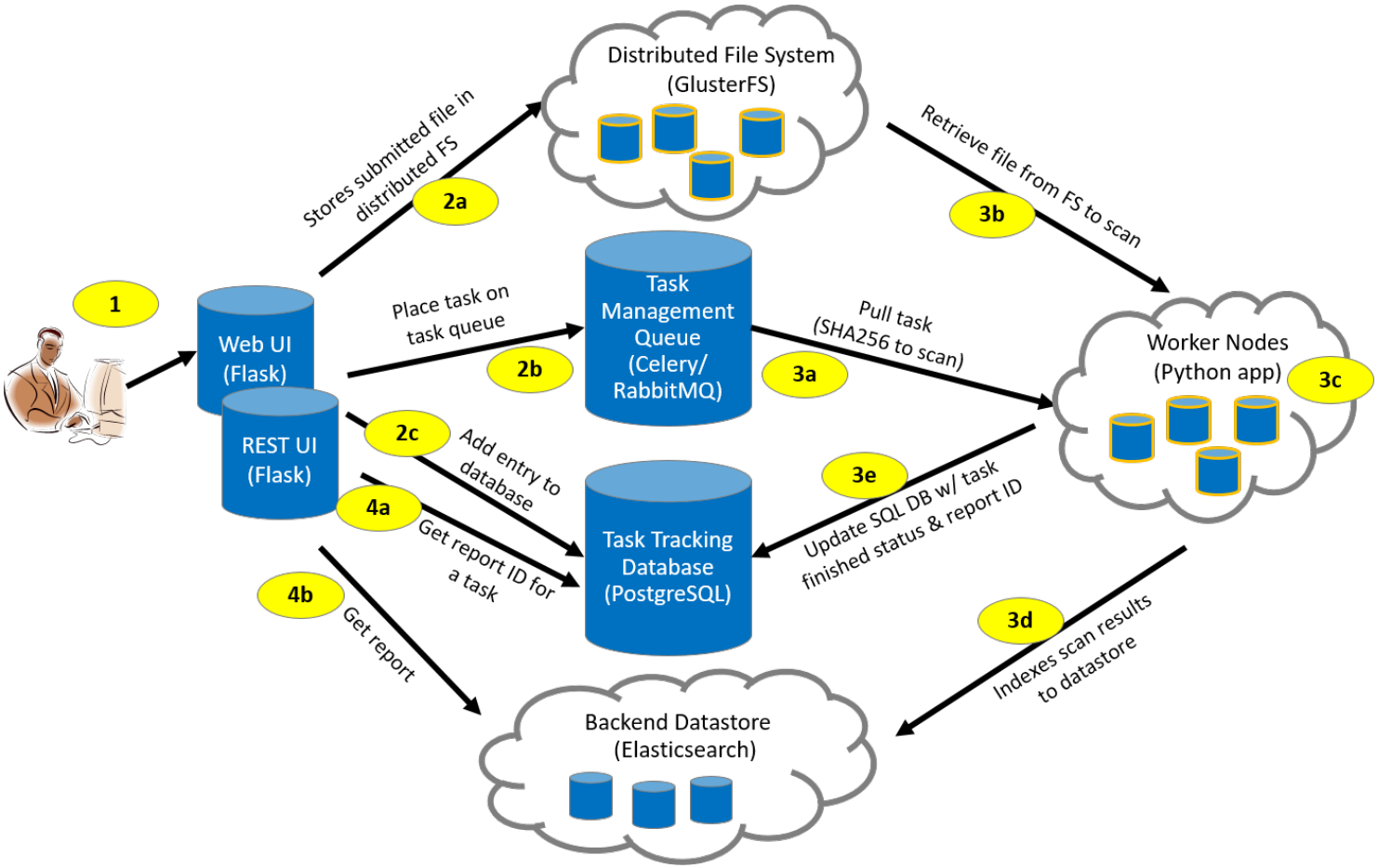
Arquitectura



MultiScanner

Introducción

Arquitectura



Varias Herramientas

Con y sin MV

Funcionalidades

HaboMalHunter

Introducción

- Es un sub-proyecto del sistema de análisis de malware Habo, el cual puede ser utilizado para automatizar el análisis de malware y evaluación de la seguridad en sistemas Linux.
- Proyecto open source (Licencia MIT). Versión 2 del 17/ene/2017.
- Funcionalidades
  - Procesar miles archivos ELF por día.
  - Integra reglas de YARA.
  - Realiza monitoreos con varias herramientas: Tcpdump.

Link

- https://github.com/Tencent/HaboMalHunter

REMnux

Introducción

- Es un sub-proyecto del sistema de análisis de malware Habo, el cual puede ser utilizado para automatizar el análisis de malware y evaluación de la seguridad en sistemas Linux.
- Proyecto open source (Licencia MIT) en donde se:
  - Descarga un OVA (VM).
  - Una distro compatible con Ubuntu.
- Funcionalidades
  - Procesar miles archivos ELF por día.
  - Integra reglas de YARA.
  - Realiza monitoreos con varias herramientas: Tcpdump.

Link

- https://remnux.org/

Angr

Introducción

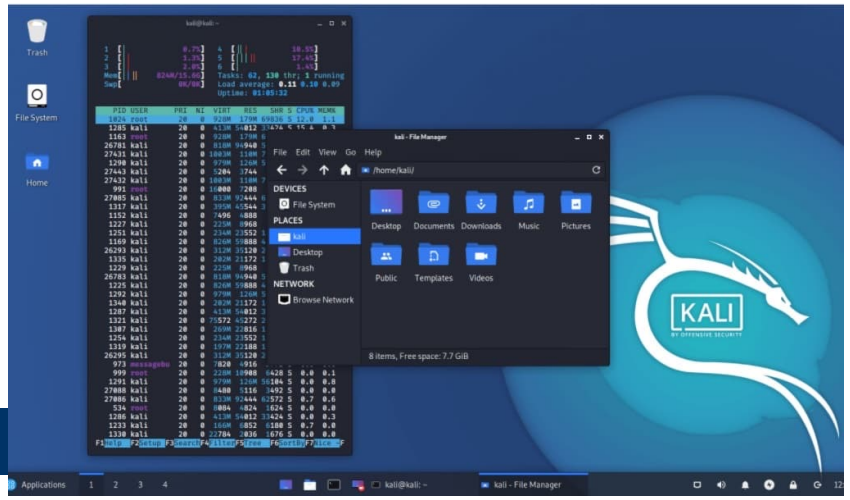
- It is a platform-agnostic binary analysis framework.
  - It is a suite of Python 3 libraries that let you load a binary and do a lot of things.
  - Disassembly, Symbolic execution, Control-flow analysis, Decompilation, etc.
  - Sin versión pero licencia del 2015.
  - Al ser Git se sabe que hay archivos de 8 años y de 3 dias.
- It is brought to you by
  - The Computer Security Lab at UC Santa Barbara
  - SEFCOM at Arizona State University, their associated CTF team.
  - Shellphish, the open source community

Link**s**

- https://github.com/angr/angr

FLARE VM

Introducción



- It is a freely available and open sourced Windows-based security distribution designed for reverse engineers, malware analysts, incident responders, forensicators, and penetration testers.
- Inspired by open-source Linux-based security distributions like:
  - Kali Linux
  - REMnux
  - and others
  - Ultima actualización diciembre del 2022.
- It also includes public malware analysis tools such as FLOSS and FakeNet-NG: empresa Mandiant.

Link**s**

- https://www.mandiant.com/resources/flare-vm-the-windows-malware

Análisis Comparativo

Caracterísiticas

Tools (comparison table)

Tabla comparativa

Funcionalidades

C0. Ambiente / Procesamiento
C1. Disassembly / Decompilation
C2. Control-flow analysis / Symbolic execution
C3. Identificar y analizar archivos
C4. Monitoreo / Volcado de memoria
C5. Yara
C6. API

Herramienta	C0	C1	C2	C3	C4	C5	C6
Cuckoo	python			✓	tcpdump Volatility		✓
MultiScanner	Colas y Tareas					✓	✓
HaboMalHunter							
REMnux	OVA, Ubuntu				tcpdump	✓	
Angr		✓ / ✓	✓ / ✓				
Mi Lab IPN							

Normalmente no da tiempo de hacer algo similar ya que son proyectos
De varios años y varias personas. Entonces se construye un
Prototipo de … Laboratorio de Malware
Con la idea de realizar un PoC (Prueba de Concepto)
Cuales son los conceptos u objetivo?

The end

Contacto

Raúl Acosta Bermejo

http:www.cic.ipn.mx
http://www.ciseg.cic.ipn.mx/

racostab@ipn.mx
racosta@cic.ipn.mx

57-29-60-00
Ext. 56652