MÁSTER EN ANÁLISIS DE MALWARE Y REVERSING

MÓDULO 2. ENTORNOS DE ANÁLISIS DE MALWARE – TAREA 1

MÁSTER EN *ANÁLISIS DE MALWARE Y REVERSING*

Campus Internacional CIBERSEGURIDAD

ENIIT INNOVA IT BUSINESS SCHOOL

UCAM Universidad Católica de Murcia

Tabla de contenido

1. INTRODUCCIÓN

Hola. Esta es la **primera evaluación** práctica de la asignatura de Entornos de Análisis de Malware.

1.1 Requisitos técnicos

Te vale cualquier sistema operativo Linux que esté soportado (es decir, que puedas instalar paquetes actualizados). Virtualiza aquel con el que prefieras trabajar. Por ejemplo, Ubuntu, Debian, Arch e incluso una Kali Linux.

Vamos a hacer uso extensivo de Python. **Recuerda que en la guía de apéndices tienes una sección dedicada a montarte tus entornos virtuales Python**. No obstante, si tienes un método mejor o con el que prefieras trabajar, adelante.

1.2 Materiales

El zip que te descargas junto a esta guía de evaluación posee la siguiente estructura:

Esos son los archivos necesarios para realizar la práctica.

1.3 Formato de entrega

Utiliza un archivo comprimido en formato zip.

Dicho archivo comprimido contendrá un documento en formato pdf con las respuestas y los archivos con el código fuente.

El nombre del archivo zip vendrá dado por:

NOMBREALUMNO_EAM_EVAL_1.zip

Dentro del comprimido, la estructura deberá ser similar a esta:

```
nombre_alumn
├── documento_entrega.pdf
├── sigma1
│   └── regla
├── snort1
│   └── regla.snort
├── snort2
│   └── regla.snort
├── yara1
│   ├── regla1.yara
│   ├── regla2.yara
│   ├── regla3.yara
│   └── regla4.yara
├── yara2
│   └── regla1.yara
└── yara3
    ├── regla1.yara
    └── regla2.yara
```

Es decir, dentro de cada carpeta irán los archivos fuentes (reglas yara, sigma, etc.) de cada ejercicio. En la raíz tu documento de entrega.

1.4 Consejos

- Por favor, escribe con buena ortografía, es necesario para hacerte comprender bien.
- Utiliza un lenguaje apropiado, profesional. Imagina que es un informe que van a leer las personas que trabajan en una empresa u organización.
- **Estas prácticas están pensadas para que empieces ya a poner a trabajar tus habilidades como reverser. Es bastante probable que tengas que encender el IDA Pro o Ghidra, un depurador o un editor hexadecimal para avanzar.**

1.5 Usa una máquina virtual desechable o con snapshots

Es importante respetar la siguiente norma:

LOS ARCHIVOS SOLO SE ABREN Y PROCESAN DENTRO DE UNA MÁQUINA VIRTUAL DESECHABLE

EVALUACIÓN

2. HASHES

2.1 Colisión

Usa los archivos de la carpeta "hashes": archivo_1 y archivo_2

Calcula el hash SHA-1 (haz captura de pantalla) y explica que está ocurriendo y por qué sucede.

¿Qué algoritmo utilizarías para obtener hashes diferentes? Explica como lo soluciona. Razona tu respuesta (añade captura de pantalla).

2.2 SSDEEP

Si usamos SSDEEP sobre **los archivos anteriores** obtenemos una salida bastante similar pero no completamente idéntica ¿Por qué está ocurriendo eso? Razona tu respuesta.

Haz captura de la salida de SSDEEP para ilustrar tu razonamiento.

3. REGLAS YARA

3.1 YARA I

Usa los archivos de la carpeta "yara1".

Escribe una regla YARA por cada archivo.

Cada una de ellas deberá detectar la presencia de uno solo de los archivos. Es decir, la regla, pongamos "regla_1.yara", detecta el file1' y no el resto. Así con las otras dos.

Escribe una cuarta regla yara que detecte dos archivos de tu elección pero no detecte el que has descartado.

En el documento pdf de entrega, **añade que estrategia has seguido para conseguirlo**, así como la elección de la sintaxis yara y tu razonamiento para encontrar la solución. Documenta con capturas de pantalla.

Recuerda:

Tus reglas se ejecutarán para evaluarte.

Además, haz capturas con la ejecución de las reglas que has hecho.

Añadir los campos de metadatos de forma correcta a tus reglas sube nota.

Recuerda añadir tus reglas al archivo comprimido.

3.2 YARA II

Usa el/los archivos de la carpeta "yara2 ".

Tenemos un archivo "sospechoso" en la carpeta "yara2" (no es malware, puedes ejecutarlo) que **se comunica con un dominio**.

Crea una regla yara que detecte dicho dominio en el ejecutable.

Cuando se te evalúe, se hará contra un ejecutable idéntico pero sin ese dominio exacto. Es decir: o creas la regla para detectar el uso del dominio exacto o no va a funcionar.

En el documento pdf de entrega, **añade que estrategia has seguido para conseguirlo**, así como la elección de la sintaxis yara y tu razonamiento para encontrar la solución. Documenta con capturas de pantalla.

Recuerda:

Tus reglas (o regla) se ejecutarán para evaluarte.

Además, haz capturas con la ejecución de las reglas que has hecho.

Añadir los campos de metadatos de forma correcta a tus reglas sube nota.

Recuerda añadir tus reglas al archivo comprimido.

3.3 YARA III

Crea una regla que detecte el archivo 1 y 3.

Crea otra regla que detecte el archivo 2 y 4.

En el documento pdf de entrega, **añade que estrategia has seguido para conseguirlo**, así como la elección de la sintaxis yara y tu razonamiento para encontrar la solución. Documenta con capturas de pantalla.

Recuerda:

Tus reglas (o regla) se ejecutarán para evaluarte.

Además, haz capturas con la ejecución de las reglas que has hecho.

Añadir los campos de metadatos de forma correcta a tus reglas sube nota.

Recuerda añadir tus reglas al archivo comprimido.

4. SANDBOXES

4.1 Sandbox I

Usando cualquier sandbox de uso gratuito tienes que encontrar una muestra que tenga estas características:

- Que sea maliciosa.
- Que tenga actividad maliciosa de red.
- Que cree archivos en el sistema.

Cuando la hayas encontrado, deberás escribir un análisis explicando cómo funciona y qué hace la muestra.

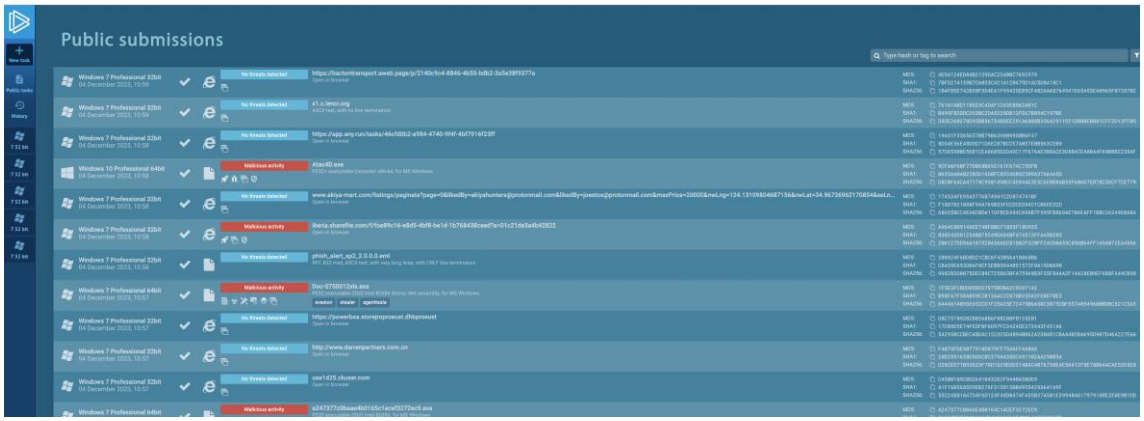
No vale comentar de forma descriptiva, como si leyeras "lo que se ve" sino tu visión y opinión analítica. Si la muestra crea decenas de archivos no es necesario que los listes sino algo similar a "la muestra crea múltiples archivos...". Interesa más que destaques algo significativo que la cantidad.

No es preciso que hagas un informe lleno de páginas. Con un par de párrafos es suficiente. Imagina que trabajas en un SOC y tienes que explicarle a un compañero o jefe qué hace la muestra.

El criterio de evaluación es precisamente tu capacidad analítica.

Usa capturas, puedes apoyarte en ellas para tu análisis.

Por ejemplo, puedes buscar entre las muestras subidas al servicio ANY.RUN o cualquier otro servicio similar:



5. REGLAS SIGMA

5.1 Sigma I

Debes recrear el entorno SIGMA que se expone en el temario y convertir una regla a tu elección que no sea la mostrada en los materiales.

Explica brevemente el proceso con tus palabras. E importante, que hace (detecta) la regla que has elegido y cómo lo hace.

Ilustra los pasos de la conversión de la regla con capturas de pantalla en tu documento de entrega PDF.

Añade la regla convertida a tu carpeta "sigma1".

6. REGLAS SNORT

6.1 SNORT I

Usa el archivo de la carpeta snort1. Es una captura de red, un archivo pcap que puedes abrir con Wireshark para inspeccionarlo si deseas (y es recomendable de cara a realizar la práctica).

En la captura de red:

0562e69fddf90f3feca3ad1b82939ffd.ngpcap se ve bastante actividad de resolución de dominios sospechosos. Concretamente, parecen dominios generados aleatoriamente (DGA).

¿Sabrías distinguir cuáles son maliciosos y cuáles no?

Crea una regla que detecte ese tipo de dominios. Fíjate como son, que características poseen.

Vamos a crear una regla que detecte este tipo de dominios. Para evaluarte se va a seguir una escala:

- Apruebas si creas una regla capaz de detectar un solo dominio.
- Apruebas y obtienes puntos extra si detectas en la misma regla todos los dominios de la captura.
- Máxima nota si detectas con la regla todos los dominios de la captura y además tu regla es capaz de detectar dominios de la misma familia (con características similares)

En tu documento de entra pdf añade capturas de la regla y la detección sobre el pcap. Añade la regla a tu carpeta "snort1".

PISTA: Lee algo sobre los DGA.

6.2 SNORT II

Usa el archivo de la carpeta snort2. Es una captura de red, un archivo pcap que puedes abrir con Wireshark para inspeccionarlo si deseas (y es recomendable de cara a realizar la práctica).

En la captura de red:

825b0080782dee075f8aac11c3a682f86c5d3aa5462bd16be0ed511a181dd7ba.ngpcap se puede ver como la máquina virtual detona una muestra que intenta comunicarse con una máquina externa.

Dichas peticiones funcionan sobre http, es fácil distinguirlas. No obstante, no resuelve a dominio alguno, sino una IP, pero no nos interesa detectar la IP puesto que en otras muestras ésta va a cambiar.

Centrémonos entonces en cómo se realiza la petición web. **Has de escribir una regla SNORT que contemple la URL que se está pidiendo**.

Recuerda que si solo detectas una porción de la URL podrías crear falsos positivos y por contra, si eres demasiado explícito solo detectaras una muestra en concreto. Intenta escribir una buena regla que equilibre ambos extremos.

Si consigues una buena regla que detecte mucho sin falsos positivos tienes puntos extra. Si tu regla solo detecta ese ejemplar concreto o potencialmente tiene muchos falsos positivos simplemente aprobarás el ejercicio.

En tu documento de entra pdf añade capturas de la regla y la detección sobre el pcap. Añade la regla a tu carpeta "snort2".

PISTA: URL también suele denominarse de forma más genérica como URI.