What is Anomal to you?

Lautaro Pinilla

Temas a presentar

1. Detección de Malware
    - Qué es la detección / qué se hace hoy en día?
2. Qué soluciones existen?
    - Comparación de alternativas
3. Presentación del Framework
    - Desarrollo de cada una de sus partes
4. Caso de uso
    - Caso detección de anomalías en DNS
5. Extensiones
    - Futuros pasos, errores conocidos

Qué es algo anómalo?

- "Irregular, extraño" - rae
- Por qué buscarlo?
  - Pueden ser indicadores de que algo no anda bien

Entonces, qué es algo anómalo?

- Lo define cada uno.
  - En base a lo que consideren **normal**

Detección de Malware

- Por qué vale la pena invertir en la detección?
  - Porque lo puedo pagar en cuotas
- Esquema Holístico vs Esquema Reactivo
  - No esperar a que pasen las cosas para mejorar

Cómo se detecta malware hoy en día

- Técnicas
- Reglas
- IDS / IA

Técnicas comunes

Hoy en día se utilizan 2 variantes de detección:

- Detección por firmas: "if ip == 8.8.8.9 then malo"
  - Reglas Yara para firmas, Reglas Sigma para comportamiento
- Detección por comportamiento: "Marcos de contaduría abrió un powershell a las 3am conectandose a turquía"

Ejemplo de regla Yara

```
rule silent_banker : banker
{
    meta:
        description = "This is just an example"
        threat_level = 3
        in_the_wild = true
    strings:
        $a = {6A 40 68 00 30 00 00 6A 14 8D 91}
        $b = {8D 4D B0 2B C1 83 C0 27 99 6A 4E 59 F7 F9}
        $c = "UVODFRYSIHLNWPEJXQZAKCBGMT"
    condition:
        $a or $b or $c
}
```

Buenas para detectar cosas ya conocidas, no sirven para técnicas novedosas

IDS / IA

IDS = Intrusion Detection System

IA = Inteligencia Artificial

Descubro comportamientos "distintos al de la mayoría"

* Cómo obtengo el "comportamiento grupal"?
* Cómo sé que no está comprometido?

Tipos de IA

- Supervizado
  - "Esto si esto no"
  - Necesitan un **dataset etiquetado** para entrenarse
  - Una vez entrenados son muy eficaces
- No Supervizado (Vamos a enfocarnos en este)
  - "Encuentra relaciones inherentes en el dataset"
  - No necesita un dataset etiquetado para entrenar
  - **Pueden producir un dataset etiquetado**

Aprendizaje No Supervizado

- Clustering
  - "Datos juntos pertenecen a un grupo"
- Graph
  - "Datos que se relacionan juntos son un mismo grupo"
- Score (ej: PCA)
  - "Datos con un score similar son un mismo grupo"

Aprendizaje No Supervizado

- **Clustering**
  - "Datos juntos pertenecen a un grupo"
- Graph
  - "Datos que se relacionan juntos son un mismo grupo"
- Score (ej: PCA)
  - "Datos con un score similar son un mismo grupo"

Mapeo en Clustering

- Cada característica que tengamos en cuenta es una dimensión
- Calculamos la distancia entre los puntos y pintamos con el mismo color los que estén cerca

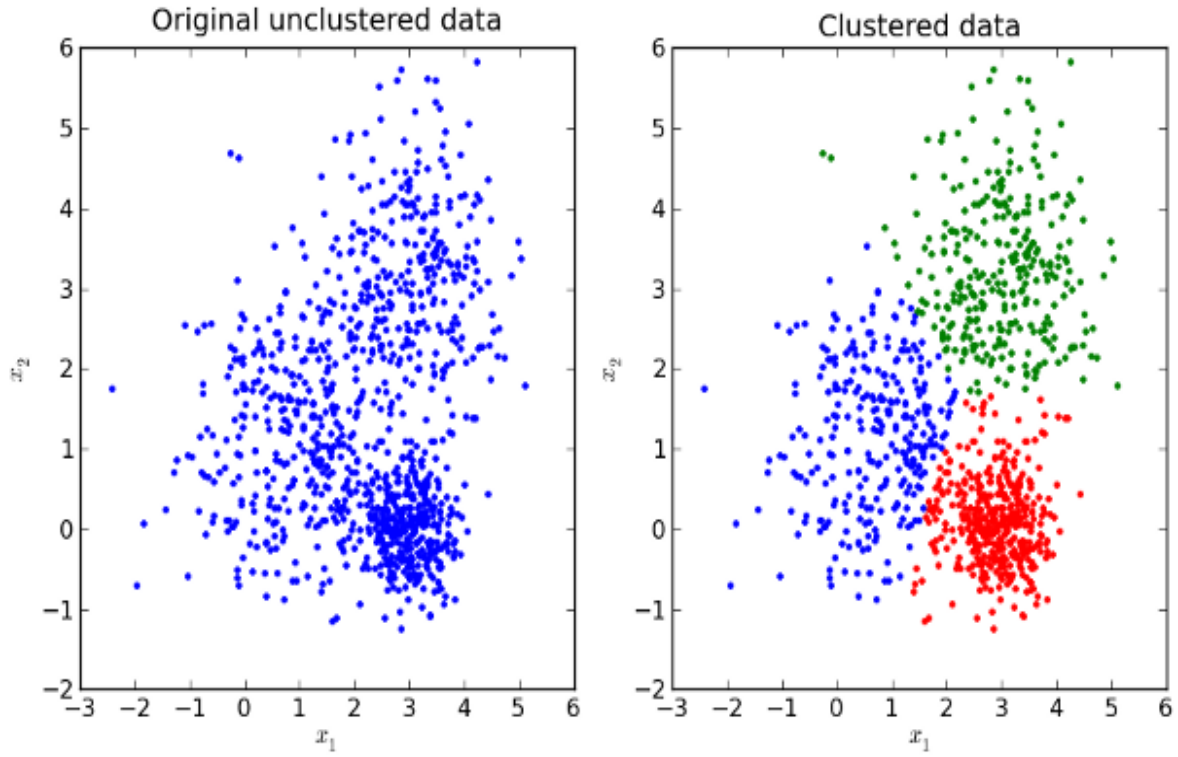


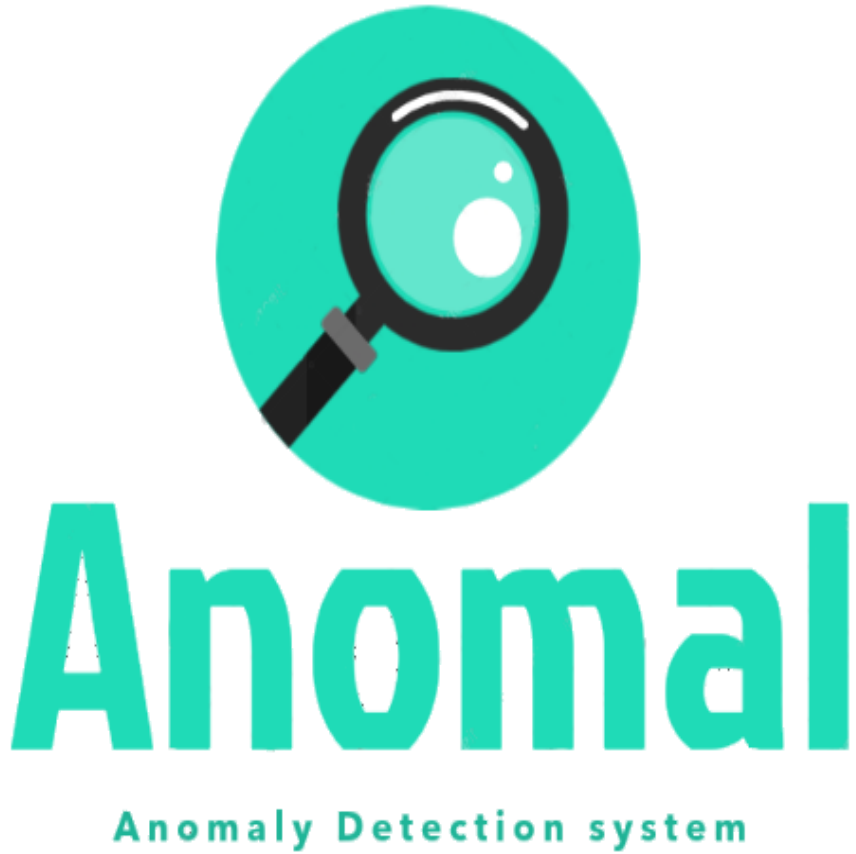
Imagen tomada de:
https://towardsdatascience.com/k-means-data-clustering-bce3335d2203

Qué soluciones existen?

Soluciones existentes

- ElasticSearch
  - Privado (licencia en USD con 75% de impuestos)
  - Limitaciones de la versión free (ej: límite de 10mb para descargar datos)
  - Detección "a oscuras"
- Apache Spark
  - Open Source
  - No tan sencillo de usar
- Stumpy
  - Open Source
  - Solo sirve para análisis en serie de tiempo

Anomal Framework



Objetivos

- Open Source
- Confiable
- Fácil de usar
- Usar tanto Rule Detection como IA
- Plug and play
- Reutilizable / Adaptable
- 2 de cappresse y 1 de humita

El Framework por dentro

El framework se descompone en Data Drivers y en 3 motores

Feature Engine

Transformar

Classification Engine

Clasificar

Report Engine

Visualizar

Cómo introduzco la data?

La data se importa y exporta con "Data Drivers". Los drivers se encargan de:

- Realizar la conexión (ya sea con un archivo csv o una db)
- Leer / Escribir la data
- Cerrar la conexión

Si en tu caso especial, tu data viene de cierta manera (json, udp, etc), se puede crear un driver para interactuar.

Feature Engine

Motor que convierte el dataset original en un dataset auxiliar.

El motor:

- Filtra los datos según los filtros definidos
- Selecciona las características elegidas
- Ejecuta las métricas basadas en las características

Ejemplo

timestamp	bytes_in	query	time_to_answer
2021-11-17T15:06:24.388Z	32.0	ekoparty.org	0.23
2021-11-17T15:06:24.385Z	28.0	sk3tchy_s1te.com	0.50

Se convierte en:

timestamp	bytes_in	query	hostname_num	base64_data
1637161584	32.0	ekoparty.org	0.23	0
1637161584	28.0	sk3tchys1t3.com/eGQK	3	1

A ver el código

```
features:
  #Fields
  - type: field
    data_needed: client.bytes
    name: client.bytes
    multiplier: 1
  - type: field
    data_needed: network.bytes
    name: network.bytes
    multiplier: 1
  #Metrics
  - type: metric
    data_needed: dns.question.name
    name: numbers_in_hostname
    multiplier: 1
  - type: metric
    data_needed: dns.question.name
    name: hostname_entropy
    multiplier: 1
```

Una métrica en código

```
def numbers_in_hostname(hostname):
    return sum(h.isnumeric() for h in hostname)
```

Flags

Son **indicadores de compromiso**.

- Corren por separado del Classification Engine
- Pueden ser "directas" o "de agregación" (el buen `groupby`)

Cómo se define un flag?

```
flags:
  - type: direct
    name: has_dns_data_b64_encoded
    data_needed: dns.answers.data
    description: This metric is used to detect dns requests with base64 encoded data.
    message: The following entries have base64 encoded data on their responses.
    severity: medium
```

Un flag en código

```
def has_dns_data_b64_encoded(dns_data):
    if dns_data is None or dns_data == '' : return False
    if dns_data[0] == '[': #list
        dns_data = dns_data[1:-1].split(';')
        for i in range(len(dns_data)):
            dns_data[i] = dns_data[i][1:-1]
    for resp in dns_data:
        if resp == '': continue
        try:
            if b64encode(b64decode(resp)).decode('ascii') == resp:
                return True
        except:
            continue
    return False
```

Cómo uso las métricas de alguien más?

Además de poder definirlos manualmente, **se pueden importar los features y métricas de un tercero**

```
plugins:
  https://github.com/userX/anomal-awesome-plugin
```

De esta forma una tercera persona puede encargarse de crear y mantener actualizadas las métricas para cierta temática.

Classification Engine

Motor que agarra el nuevo dataset y produce una clasificación de los datos

- Cómo?
  - Como se lo haya configurado (aunque hay un motor default)

Volviendo a las Features

```
  - type: metric
    data_needed: dns.question.name
    name: hostname_entropy
    multiplier: 1
```

multiplier?

Clasificación con métrica de Gower

Preguntas comunes:

1. Por qué no usas kmeans?
    - Porque las métricas pueden dar resultados categóricos
    - Puedo asignarles un peso a cada feature / métrica que tengo
2. Qué es un gower?
    - Una métrica / distancia para calcular qué tan cerca están dos puntos
3. Existe el token GowerInu?
    - Probablemente pero por las dudas no inviertas ahí

Cómo funciona la métrica?

Math Warning:

- La función `s(x1,x2)` cambia de acuerdo al tipo de variables.
- Para descriptores qualitativos, se utiliza la distancia `Dice`.

$$D_{Gower}(x_1, x_2) = 1 - \left(\frac{1}{p} \sum_{j=1}^{p} s_j(x_1, x_2) \right)$$

Imagen de https://medium.com/analytics-vidhya/gowers-distance-899f9c4bd553

Yo quería Kmeans..

El Classification Engine es **intercambiable**. Uno puede elegir **el método de clasificación que más le convenga** sin afectar al resto del sistema.

Report Engine

Se encarga de mostrar los resultados para su estudio y análisis

El default es una página web pero se puede crear un nuevo Engine si se quiere mostrar la data como pdf, json, etc.

Contiene:

- El dataset sospechoso
- Información de las features para su estudio
- Sección aparte para las flags

Caso de uso

Detección de malware en red basándose en actividad DNS

Detección de malware en red basándose en actividad DNS

- Objetivo
  - Solo mirando data DNS, encontrar malware en una red
- Por qué DNS?
  - Porque los atacantes lo usan y los defensores lo suelen ignorar
- Es data sensible?
  - Si, con esto se puede ver **TODO**. No importa si hay tráfico encriptado o no, vas a tener que resolver nombre -> IP

Ejemplo de consulta DNS

```
; <<>> DiG 9.16.1-Ubuntu <<>> ekoparty.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 12862
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;ekoparty.org.                  IN      A

;; ANSWER SECTION:
ekoparty.org.           51      IN      A       138.68.21.56

;; Query time: 12 msec
;; SERVER: 192.168.0.1#53(192.168.0.1)
;; WHEN: vie sep 23 12:26:13 -03 2022
;; MSG SIZE  rcvd: 57
```

Qué técnicas de ataques queremos detectar?

- Tunneling
  - Utilizar el protocol DNS como "túnel" para otros protocolos
- APT
  - Comunicación entre un agente y su C2
- DGA
  - Dominios generados aleatoriamente

Qué métricas podemos utilizar

Métrica	Puede detectar	Necesito
Cantidad de Requests y Response por IP	Tunneling	Request + Response
Geolocalización de la IP	Tunneling	Hostname
Consultas DNS Sin Response	Tunneling	Request + Response
Cantidad de números en el hostname	DGA	Hostname
Patrón de frecuencias	C2	Request + Response
Reputación de un dominio	C2	Hostname

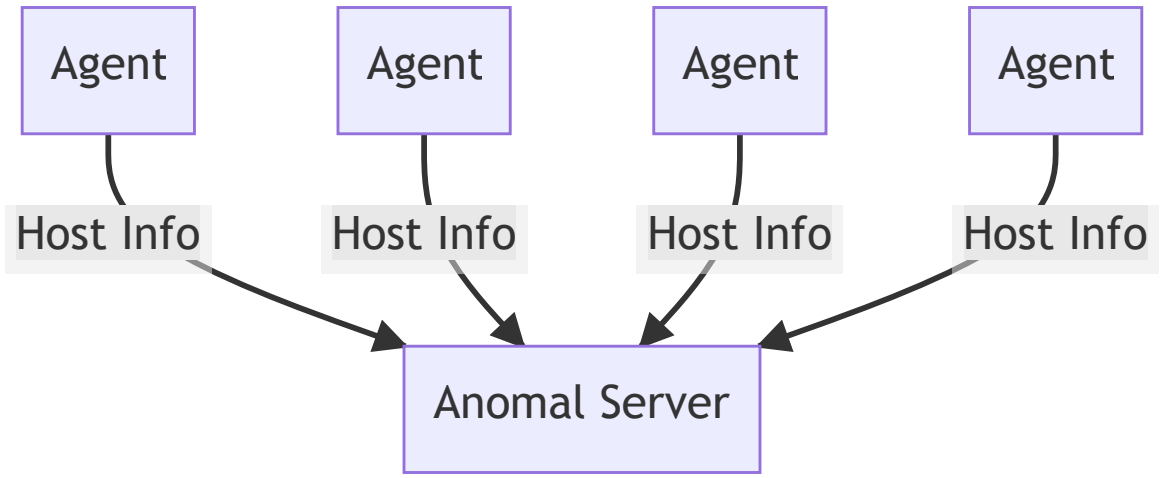
DNS Features Parte 2

Métrica	Puede detectar	Necesito
Reputación de un dominio	C2	Hostname
Tiempo en que tarda en volver la consulta	APT	Request + Response
Consultas Vacías	APT	Requests
TTL No Estandar	APT	Response
Base64 Analysis	APT	Hostname

Mini Demo

Futuras extensiones

- Múltiples plugins al mismo tiempo
- Soporte para análisis en tiempo real (en desarrollo)
- Serverless



Modo Tiempo Real

Tener "agentes" corriendo en los equipos que recolectan data de este y lo envían al servidor (low-budget nessus)

Arquitectura Serverless

Que cada feature / métrica se procese por separado y después se junten utilizando arq serverless.

Reduciría los costos ya que pagaríamos por cómputo en vez de por instancia

Errores conocidos

- Los gráficos de features pueden presentar errores
- El framework crasheó con grandes volúmenes de datos, falta testear más para tener un límite aproximado

Conclusiones

1. Aprendimos un poco sobre detección de malware
2. Vimos distintas técnicas para lograrlo
3. Se presentó el framework Anomal

Muchas gracias

Gracias Ekoparty!

Preguntas?