# HoneyPot Industriel

Vincent ROME - Axel FOULON - Julien DUPAGNY / Master CDSI 2017-2018

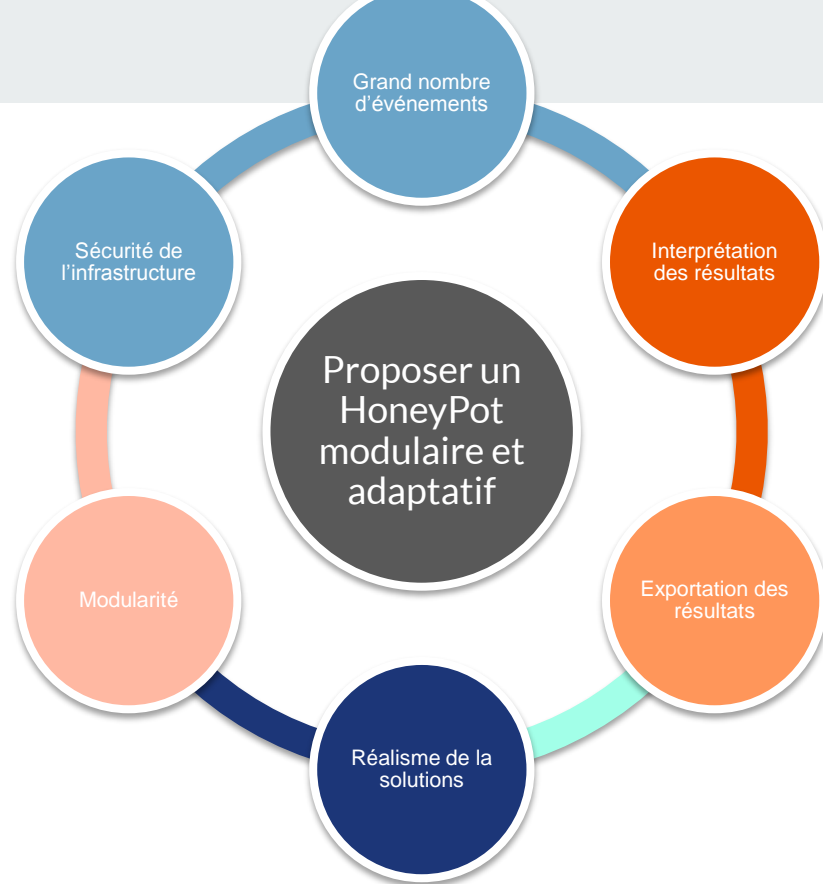# Sommaire

**Principe HoneyPot**

**Le protocole Modbus**
- Recherche de bannières
- Librairie Pymodbus

**Architecture**
- Technologies utilisées
- Déploiement de l'architecture

**Démonstration**

# Sommaire



Scripts d'attaques

Améliorations

Conclusion
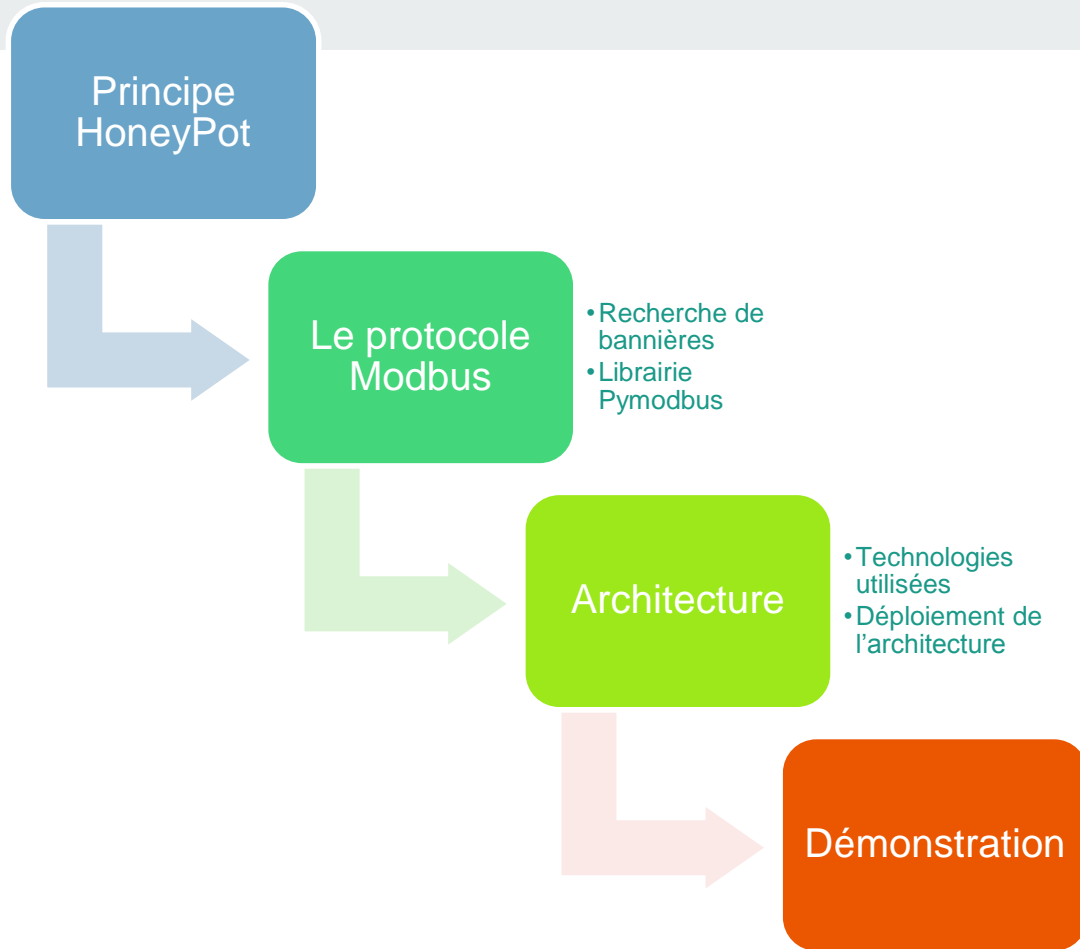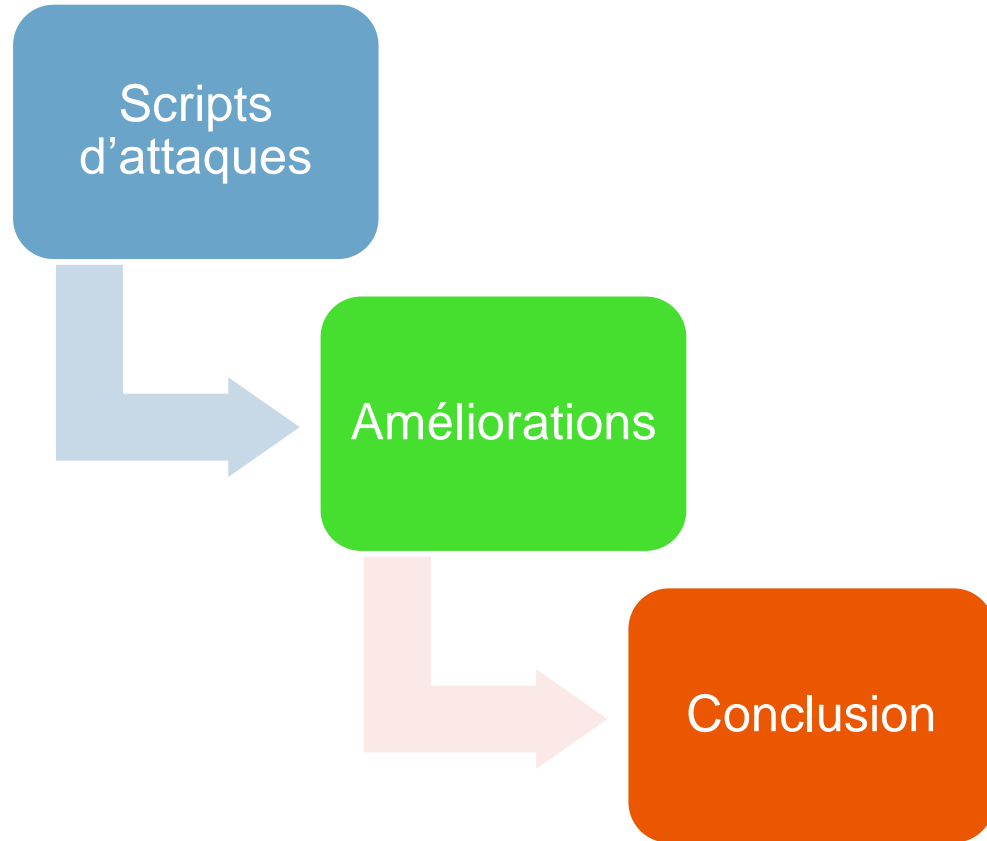
# Objectif et contraintes

100 heures hommes en scolaire
60 heures hommes en personnel

Grand nombre d'événements

Sécurité de l'infrastructure

Interprétation des résultats

Proposer un HoneyPot modulaire et adaptatif

Modularité

Exportation des résultats

Réalisme de la solutions

# Principe HoneyPot

# Caractéristiques Modbus



Utilisation : Utilisé dans le monde industriel

Modes : RTU (série) / TCP / UDP

Architecture : Client/Serveur

CRC : Géré parRS ou TCP/IP

Aucune sécurité embarquée
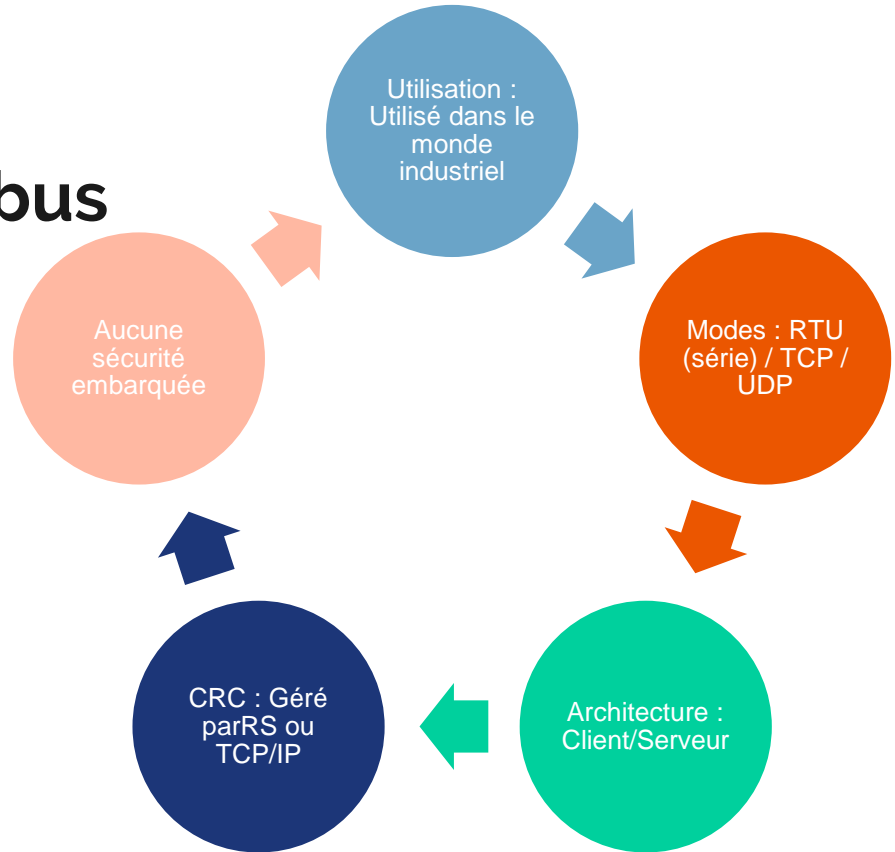
# Caractéristiques Modbus

| Bloc de mémoire | Type de données | Accès par le maître | Accès par l'esclave |
|---|---|---|---|
| Bobines | Booléen | Lire/Écrire | Lire/Écrire |
| Entrées discrètes | Booléen | Lecture seule | Lire/Écrire |
| Registres de maintien | Mot non signé | Lire/Écrire | Lire/Écrire |
| Registres d'entrée | Mot non signé | Lecture seule | Lire/Écrire |

# Caractéristiques Modbus



Nombre d'octets : 8  6  6  2  20  20  46 à 1500  4

Préambule | MAC Destin. | MAC Source | LLC (Type) | IP | TCP | Applications | Contrôle FCS

Function Code | Function-Specific Data
0  1  253 max

En-tête Ethernet ← → Données ← → CRC
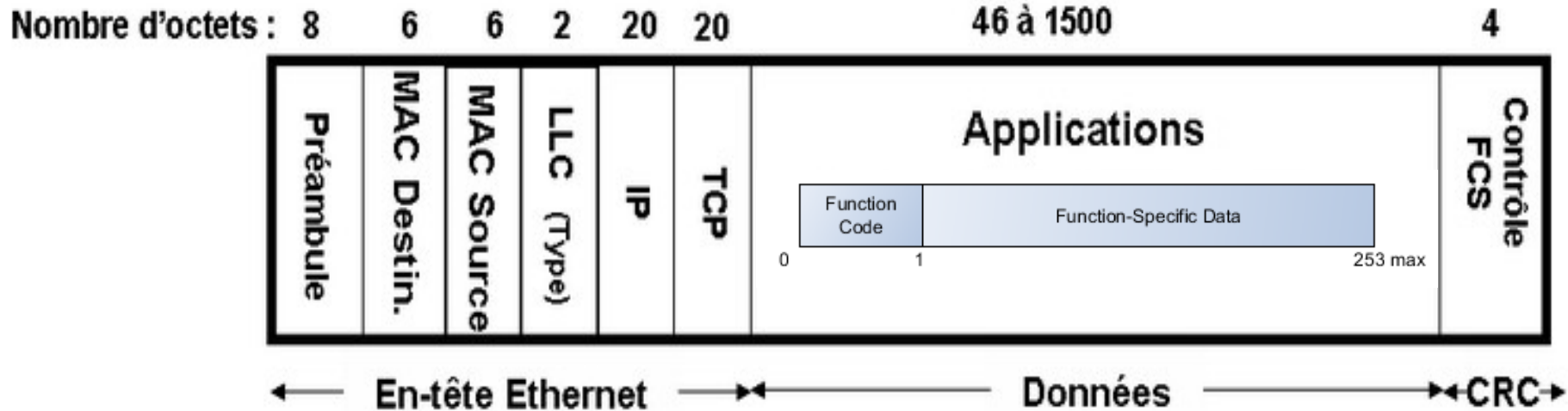
# Caractéristiques Modbus

| Code | Description |
|------|-------------|
| 3 | Lire des registres multiples |
| 16 | Écrire des registres multiples |

# Caractéristiques Modbus

| Code | Description |
|------|-------------|
| 1 | Lire des bobines |
| 2 | Lire des entrées discrètes |
| 4 | Lire des registres d'entrées |
| 5 | Écrire une bobine simple |
| 6 | Écrire un registre simple |
| 7 | Lire un statut d'exception (série uniquement) |

# Caractéristiques Modbus

| Code | Description |
|------|-------------|
| 15 | Écrire des bobines multiples |
| 20 | Lire un enregistrement de fichier |
| 21 | Écrire un enregistrement de fichier |
| 22 | Masquer Lire un registre |
| 23 | Lire/Écrire des registres multiples |
| 24 | Lire FIFO |

# Recherche de bannières
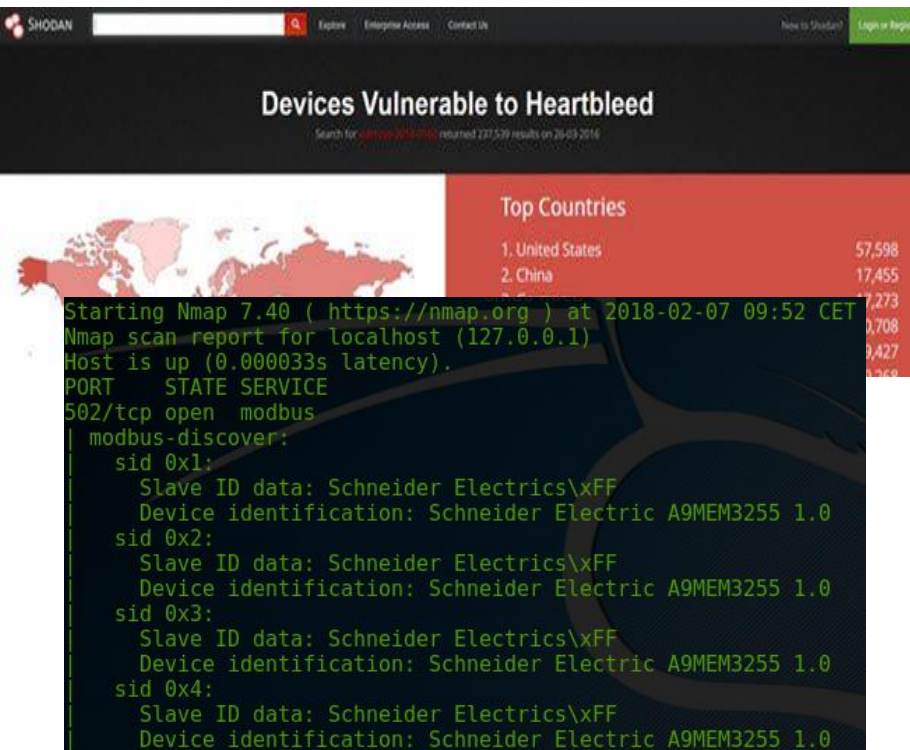


| TOP PRODUCTS |
|---|
| (1) - BMX P34 2020   566<br>(2) - BMX NOE 0100   230<br>(3) - TM221CE40T     174<br>(4) - SAS TSXETY4103 155<br>(5) - TM221CE40R     57 |
| TYPICAL OUTPUT |
| Unit ID: 0<br>-- Slave ID Data:     @ X<br>             (0808ffe14000580000000000)<br>-- Device Identification: Schneider Electric A9MEM3255 V1.0 |

# Librairie Pymodbus

Création de Serveurs et De client

Implémentation du protocole en python

Logging peu adapté, non mis en forme et bavard
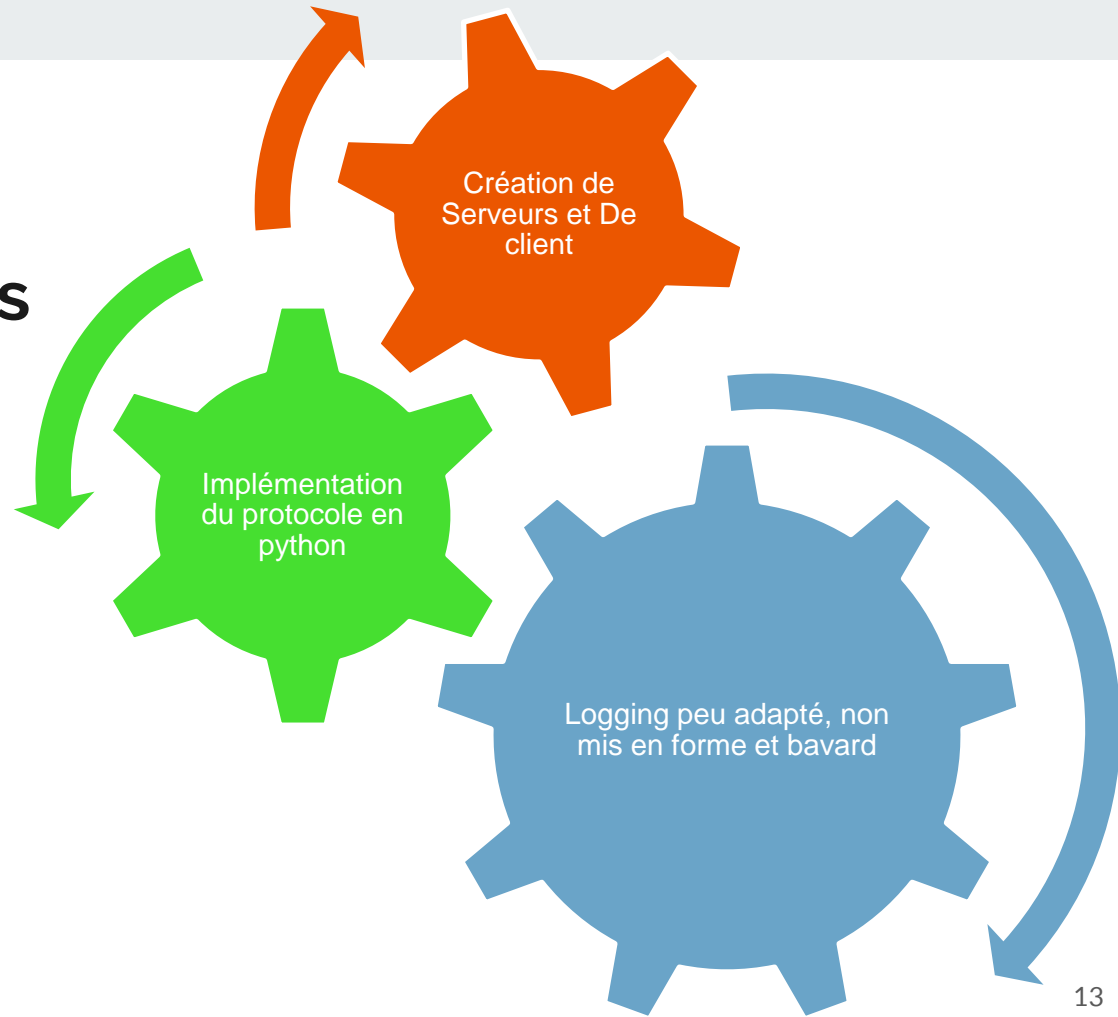
```python
def _helper(self, data, client_address=None):
    '''
    This factory is used to generate the correct request object
    from a valid request packet. This decodes from a list of the
    currently implemented request types.

    :param data: The request packet to decode
    :returns: The decoded request or illegal function request object
    '''
    function_code = byte2int(data[0])
    if client_address != None :
        extra = {
                'ip_source'   : client_address[0],
                'port_source' : client_address[1],
                'fonction_code' : function_code,
                'requete' : str(data)
            }
    else :
        extra = {
                'fonction_code' : function_code,
                'requete' : str(data)
            }
    #_logger.debug("test:"+ str(data))
    _logger.debug("traité " + str(extra),extra=extra)
```

14

# Exemple De Log

```
Wed Feb  7 09:51:52 2018 machinedetest  python-logstash-logger DEBUG Client Connected {'port_source': 51108, 'ip_source': '192.168.2.10'}
Wed Feb  7 09:51:52 2018 machinedetest  python-logstash-logger DEBUG traité {'requete': "b'\\x11'", 'port_source': 51108, 'fonction_code': 17,
'ip_source': '192.168.2.10'}
Wed Feb  7 09:51:52 2018 machinedetest  python-logstash-logger DEBUG Client Connected {'port_source': 51110, 'ip_source': '192.168.2.10'}
Wed Feb  7 09:51:52 2018 machinedetest  python-logstash-logger DEBUG traité {'requete': "b'+\\x0e\\x01\\x00'", 'port_source': 51110, 'fonction_code': 43
, 'ip_source': '192.168.2.10'}
Wed Feb  7 09:51:52 2018 machinedetest  python-logstash-logger DEBUG Client Connected {'port_source': 51112, 'ip_source': '192.168.2.10'}
Wed Feb  7 09:51:52 2018 machinedetest  python-logstash-logger DEBUG traité {'requete': "b'\\x11'", 'port_source': 51112, 'fonction_code': 17,
```

# Docker

Isolation des applications

Portabilité

Docker Swarm ( Cluster)

Scalabilité

| App 1 | App 2 | App 3 |
|---|---|---|
| Bins/Libs | Bins/Libs | Bins/Libs |
| Guest OS | Guest OS | Guest OS |

Hypervisor

Host Operating System

Infrastructure

**Virtual Machines**

| App 1 | App 2 | App 3 |
|---|---|---|
| Bins/Libs | Bins/Libs | Bins/Libs |

Docker Engine

Operating System

Infrastructure
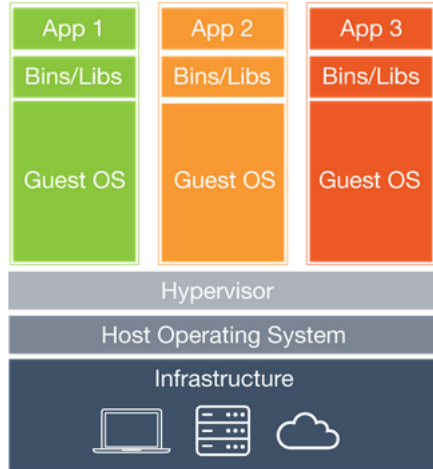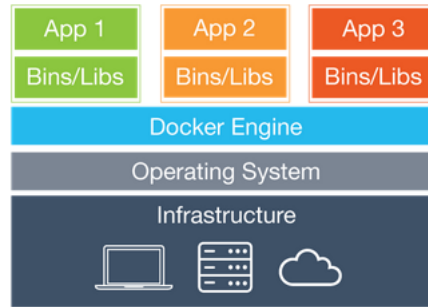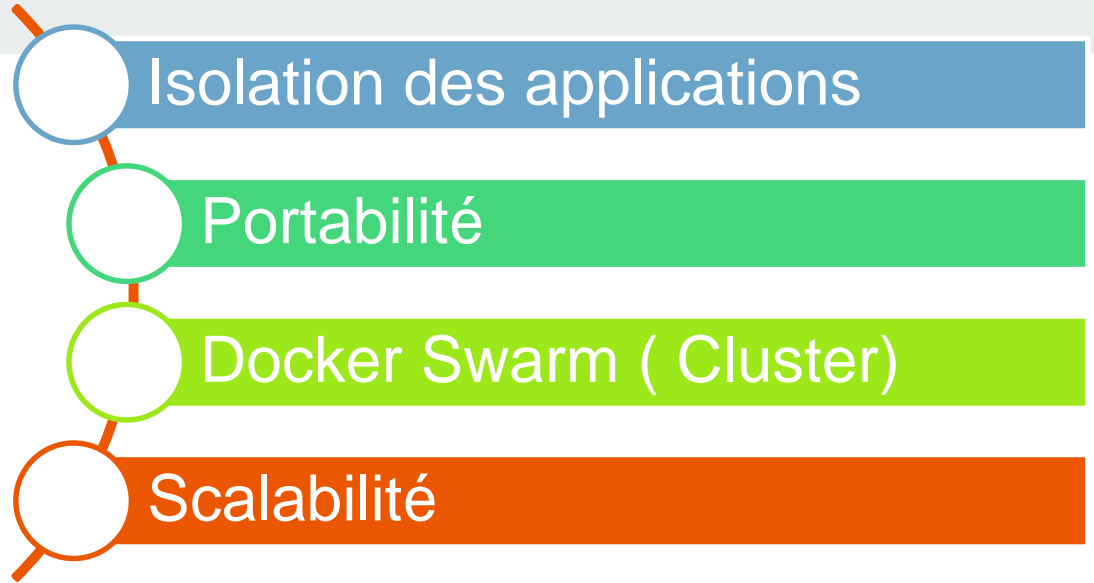
**Containers**

# ELK Stack

LOG → logstash → elasticsearch → kibana

## x-pack

| Security | Alerting | Monitoring | Reporting | Graph | Machine Learning |
|----------|----------|------------|-----------|-------|------------------|
| (anciennement Shield) | (via Watcher) | (anciennement Marvel) | | | |

## beats
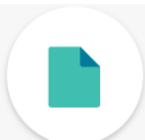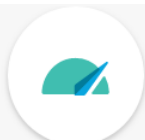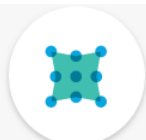
| Filebeat | Metricbeat | Packetbeat | Winlogbeat | Auditbeat | Heartbeat |
|----------|------------|------------|------------|-----------|-----------|
| Logs fichiers | Indicateurs | Données réseau | Logs Windows | Audit Data | Uptime Monitoring |

https://www.elastic.co/fr/subscriptions
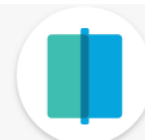
| | OPEN SOURCE | BASIC |
| --- | --- | --- |
| | Téléchargement gratuit | Licence gratuite |
| **SUITE ELASTIC / ELASTIC STACK** | | |
| Elasticsearch | ✓ | ✓ |
| Kibana | ✓ | ✓ |
| Beats | ✓ | ✓ |
| Logstash | ✓ | ✓ |
| **X-PACK** | | |
| ⌄ Security (anciennement Shield) | | |
| ⌃ Monitoring (anciennement Marvel) | | ✓ |
|    Monitoring full-stack | | ✓ |
|    Prise en charge du monitoring multi-stack | | |
|    Politique de rétention configurable | | |
|    Alertes automatiques en cas de problèmes de la stack | | |
| ⌃ Management | | ✓ |
|    API et UI d'assistant de mis à niveau | | ✓ |
|    Gestion des pipeline Logstash | | |
| ⌄ Alerting (via Watcher) | | |

| | |
| --- | --- |
| ⌄ Alerting (via Watcher) | |
| ⌄ Machine Learning | |
| ⌄ Graph Analytiques & Visualisation | |
| Reporting | ✓ |
|    Export au format CSV | ✓ |
|    Export au format PDF | |
| ⌃ Modules | ✓ |
|    Module Logstash ArcSight | ✓ |
| ⌃ Dev Tools | ✓ |
|    Search Profiler | ✓ |
|    Grok Debugger | ✓ |

18

# Notre architecture

# Création des dockers
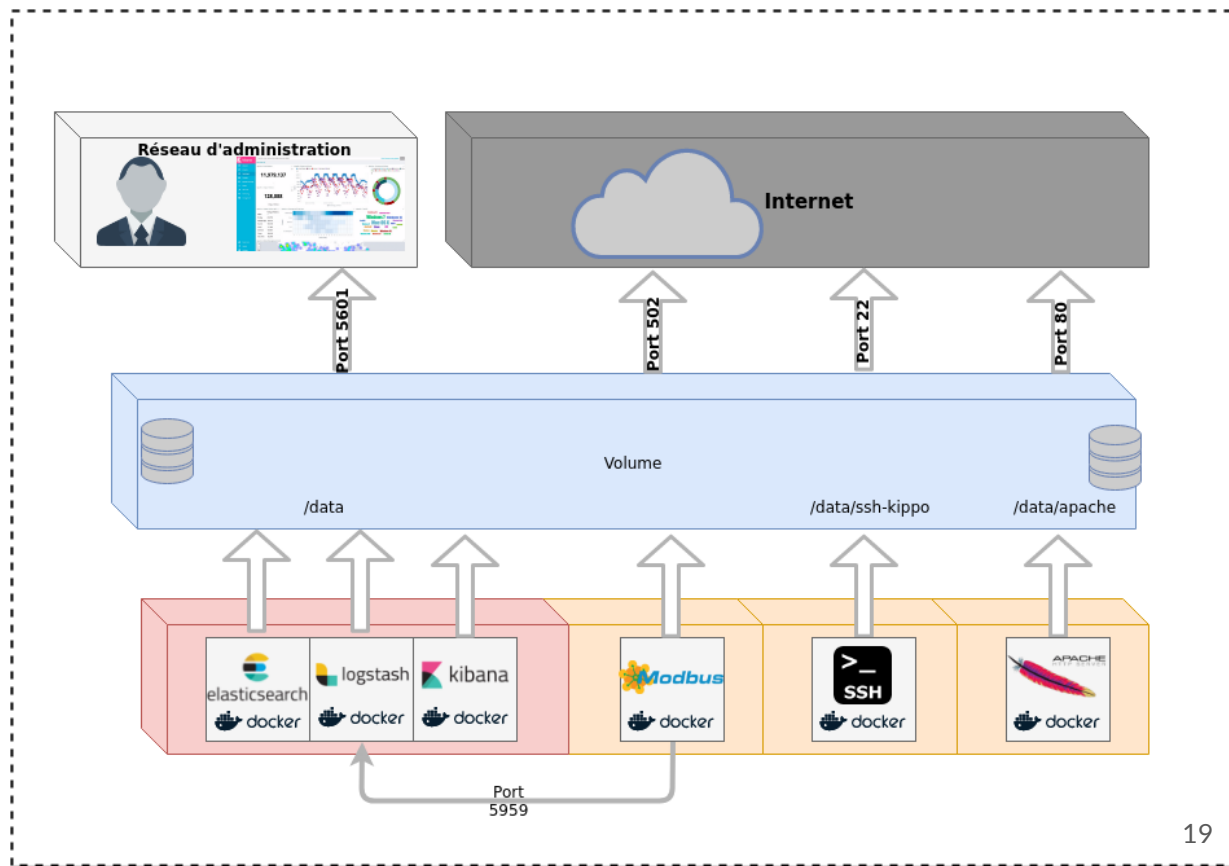


https://github.com/deviantony/docker-elk
- Modification des configurations

- Création de 3 Dockerfiles

# Dockerfile



```
FROM debian:latest
EXPOSE 502/tcp
RUN apt-get update && \
    apt-get upgrade -y && \
    apt-get install -y \
    python3 \
    python3-dev \
    python3-pip  \
    git \
    python3-setuptools \
    virtualenv && \
    apt-get autoremove && \
    apt-get clean && \
    pip3 install -U python-logstash
RUN git clone https://github.com/lightoyou/HoneyPot_CDSI_pymodbus.git
RUN cd HoneyPot_CDSI_pymodbus && python3 setup.py install
COPY ./server.py /root
CMD ["python3","/root/server.py" "0"]
```

**Dockerfile Modbus**

# Résultats

# Résultats



```
PORT     STATE SERVICE VERSION
22/tcp  open  ssh      OpenSSH 5.1p1 Debian 5 (protocol 2.0)
|_banner: SSH-2.0-OpenSSH_5.1p1 Debian-5
80/tcp  open  http     Apache httpd 2.4.10 ((Debian))
|_http-server-header: Apache/2.4.10 (Debian)
502/tcp open  mbap?
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

# Gestion des "logs"

# Formatage dans Logstash

```
input {
  #Modbus
  tcp {
    port => 5959
    codec => json
  }
  #apache
  file {
    path => ["/data/apache/access.log"]
    type => "apache_access"
  }
  file {
    path => ["/data/apache/error.log"]
    type => "apache_error"
  }
  file {
    path => ["/data/ssh_kippo/kippo.log"]
    type => "kippo"
  }
}
```

```
filter {
  if [type] in [ "apache" , "apache_access" , "apache-access" ]  {
    grok {
      match => [
        "message" , "%{COMBINEDAPACHELOG}+%{GREEDYDATA:extra_fields}",
        "message" , "%{COMMONAPACHELOG}+%{GREEDYDATA:extra_fields}"
      ]
      overwrite => [ "message" ]
    }
    mutate {
      convert => ["response", "integer"]
      convert => ["bytes", "integer"]
      convert => ["responsetime", "float"]
    }
```

Search... (e.g. status:200 AND extension:PHP)   Uses lucene query syntax

Add a filter +

\* 

Selected Fields

? _source

Available Fields ⚙

t @metdata.ip_address

⏱ @timestamp

t @version

t _id

t _index

# _score

t _type

? agent

? auth

? build

? bytes

? clientip

t day

t dayname

t device

? dstip

? dstport

t host

**_source**

► **message:** 2018-02-06 17:56:59+0000 [kippo.core.ssh.HoneyPotSSHFactory] New connection: 172.23.0.1:59398 (172.23.0.3:2222) [session: 4] **time:** 17:56:59 **srcip:** 172.23.0.1 **month:** 02 **type:** kippo **dstport:** 2222 **path:** /data/ssh_kippo/kippo.log **srcport:** 59398 **dstip:** 172.23.0.3 **kippo-type:** connection **day:** 06 **kippo-session:** 4 **host:** 039a26e1dd0c **@timestamp:** February 6th 2018, 18:56:59.000 **timestamp:** 2018-02-06 17:56:59 **@version:** 1 **year:** 2018 **_id:** IkBDbGEBFBo5Sa1LhFpK **_type:** system_logs **_index:** apache-2018.02.06 **_score:** -

► **message:** 2018-02-06 17:56:59+0000 [kippo.core.ssh.HoneyPotSSHFactory] New connection: 172.23.0.1:33110 (172.23.0.3:2222) [session: 5] **time:** 17:56:59 **srcip:** 172.23.0.1 **month:** 02 **type:** kippo **dstport:** 2222 **path:** /data/ssh_kippo/kippo.log **srcport:** 33110 **dstip:** 172.23.0.3 **kippo-type:** connection **day:** 06 **kippo-session:** 5 **host:** 039a26e1dd0c **@timestamp:** February 6th 2018, 18:56:59.000 **timestamp:** 2018-02-06 17:56:59 **@version:** 1 **year:** 2018 **_id:** IUBDbGEBFBo5Sa1LhFpI **_type:** system_logs **_index:** apache-2018.02.06 **_score:** -

► **message:** [pid 16:tid 140446208358144] [client 172.22.0.1:60010] AH01630: client denied by server configuration: /var/www/public-html/server-status **time_stamp:** 06/Feb/2018:17:55:45.18985 1 **month:** Feb **loglevel:** authz_core:error **hour:** 17 **type:** apache_error **path:** /data/apache/error.log **minute:** 55 **dayname:** Tue **day:** 06 **second:** 45.189851 **host:** 039a26e1dd0c **@timestamp:** February 6th 2018, 18:55:45.396 **tags:** _dateparsefailure **@version:** 1 **year:** 2018 **_id:** H0BCbGEBFBo5Sa1LYlqF **_type:** system_logs **_index:** apache-2018.02.06 **_score:** -

► **device:** Other **auth:** - **request:** /server-status **type:** apache_access **path:** /data/apache/access.log **ident:** - **httpversion:** 1.1 **tags:** _geoip_lookup_failure **os_name:** Other **message:** 17 2.22.0.1 - - [06/Feb/2018:17:55:45 +0000] "GET /server-status HTTP/1.1" 403 477 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)" **response:** 403 **clientip:** 172.22.0.1 **name:** Other **os:** Other **build:** **bytes:** 477 **verb:** GET **referrer:** "-" **agent:** "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)" **host:** 039a26e1dd0c **@timestamp:** February 6th 2018, 18:55:45.000 **@version:** 1 **_id:** IEBCbGEBFBo5Sa1LZVqK **_type:** system_logs **_index:** apache-2018.02.06 **_score:** -

► **message:** [pid 16:tid 140445975774976] [client 172.22.0.1:60004] AH01630: client denied by server configuration: /var/www/public-html/server-status **time_stamp:** 06/Feb/2018:17:55:32.91436 6 **month:** Feb **loglevel:** authz_core:error **hour:** 17 **type:** apache_error **path:** /data/apache/error.log **minute:** 55 **dayname:** Tue **day:** 06 **second:** 32.914366 **host:** 039a26e1dd0c **@timestamp:** February 6th 2018, 18:55:33.389 **tags:** _dateparsefailure **@version:** 1 **year:** 2018 **_id:** HkBCbGEBFBo5Sa1LM1qg **_type:** system_logs **_index:** apache-2018.02.06 **_score:** -

► **device:** Other **auth:** - **request:** /server-status **type:** apache_access **path:** /data/apache/access.log **ident:** - **httpversion:** 1.1 **tags:** _geoip_lookup_failure **os_name:** Other **message:** 17 2.22.0.1 - - [06/Feb/2018:17:55:32 +0000] "GET /server-status HTTP/1.1" 403 477 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)" **response:** 403 **clientip:** 172.22.0.1 **name:** Other **os:** Other **build:** **bytes:** 477 **verb:** GET **referrer:** "-" **agent:** "Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)" **host:** 039a26e1dd0c **@timestamp:** February 6th 2018, 18:55:32.000 **@version:** 1 **_id:** HUBCbGEBFBo5Sa1LMlq7 **_type:** system_logs **_index:** apache-2018.02.06 **_score:** -

► **message:** **port:** 34,490 **type:** logstash **logger_name:** python-logstash-logger **path:** /usr/local/lib/python3.5/dist-packages/pymodbus-1.4.0-py3.5.egg/pymodbus/server/sync.py **stack_info:** - **level:** DEBUG **host:** 99503947eff7 **@timestamp:** February 6th 2018, 18:54:11.121 **tags:** **@version:** 1 **@metdata.ip_address:** 172.21.0.3 **_id:** GkBAbGEBFBo5Sa1L81rP **_type:** system_logs **_index:** apache-2018.02.06 **_score:** -

26

# Création de visualiser et dashboard

# DÉMONSTRATION

# Scripts d'attaques - Brute-force ssh

- Attaque par dictionnaire
- Essaie de plusieurs combinaisons
- Informe lorsque les identifiants sont trouvés



*Output du script*

# Scripts d'attaques - Déni de service

- Ouvre une infinité de connexion sur le PLC



*Output du serveur*



*Output du script*

# Scripts d'attaques - Écriture aléatoire

- Écriture dans les registres (random)
- Lecture dans les registres pour vérifier les valeurs



*Output du script*

# Scripts d'attaques - Écriture aléatoire



*Output du serveur*

Global Traffic

Modbus-Activity

Web-Activity-by-OS
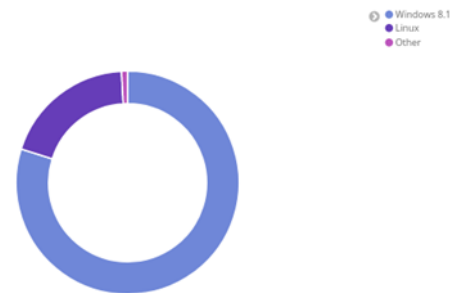
**Modbus-Activity-by-IP**

| IP ⇕ | Nombre de connexions ⇕ |
|---|---|
| 192.168.1.32 | 20,868 |
| 192.168.2.10 | 4,066 |
| 192.168.1.62 | 784 |
| | **25,718** |

Export: Raw ⬇  Formatted ⬇

**SSH-Activity-by-IP**

| IP ⇕ | Nombre de connexions ⇕ |
|---|---|
| 192.168.1.32 | 73 |
| 192.168.2.10 | 30 |
| 192.168.2.1 | 12 |
| | **126** |

Export: Raw ⬇  Formatted ⬇

**Web-Activity-by-IP**

| IP ⇕ | Nombre de connexions ⇕ |
|---|---|
| 192.168.1.32 | 354 |
| 192.168.1.62 | 82 |
| 192.168.2.10 | 11 |
| | **457** |

Export: Raw ⬇  Formatted ⬇

**Modbus-Ecriture-moyenne**

● Ecritures

**Pymodbus Most Used Fonctions**

| Fonction ⇕ | Nombre ⇕ |
|---|---|
| 3 | 4,747 |
| 1 | 4,680 |
| 15 | 2,345 |
| 5 | 2,296 |
| 2 | 2,278 |

Export: Raw ⬇  Formatted ⬇

**SSH-Most-Used-Password**

● Count

34

# Amélioration

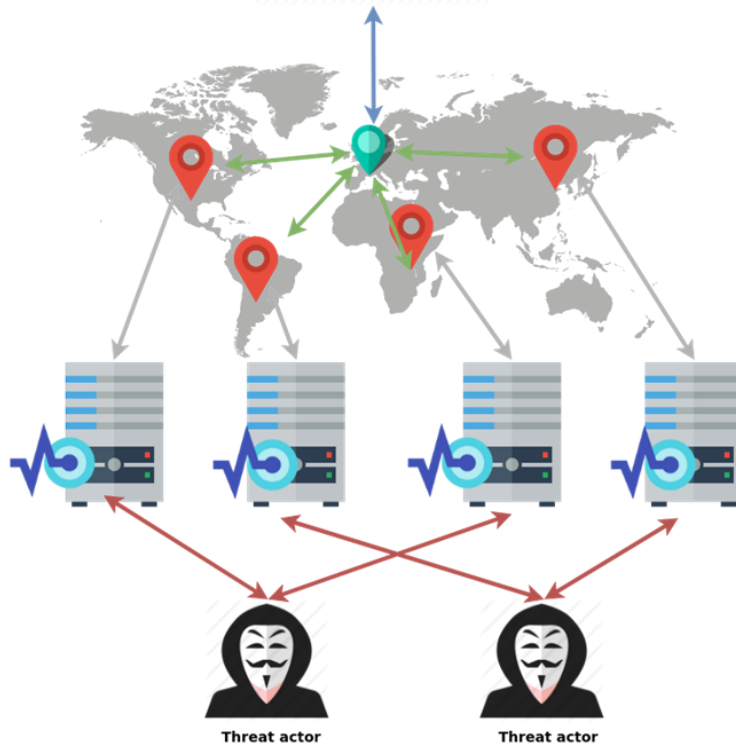Améliorations → Sécurisation ELK → Amélioration des IP tables → Programme d'installation → Ajouter des Dashboard → Supervision Docker
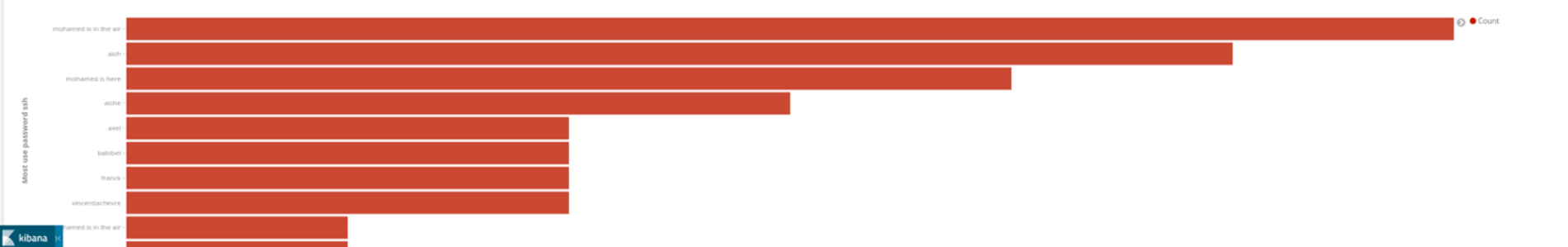
# Un serveur/ Des serveurs

- Permettre le monitoring des sondes
- Permettre le déploiement facile de sondes

## Nous
(Utilisateur)

11,979,137

128,088

# Partage communauté

- Echanger des observables
- Avoir des meilleurs statistique

**HN/P**

# Sondes :

Une variété d'HoneyPot
(Exploit Target)

Threat actor          Threat actor

THREATSTREAM®

MHN

mongoDB

Mnemosyne

hpfeeds-logger

hpfeeds → honeymap    Webapp    REST API

logstash

Sensors

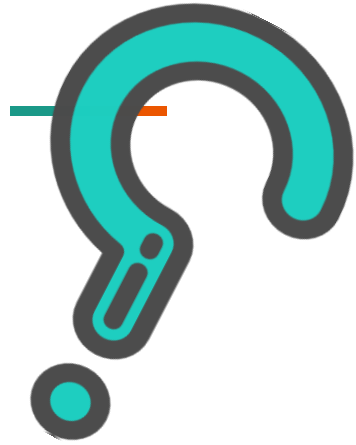| kippo | snort | conpot | dionaea | YOURS |
| kippo | snort | conpot | dionaea | YOURS |
| kippo | snort | conpot | dionaea | YOURS |

3rd party apps
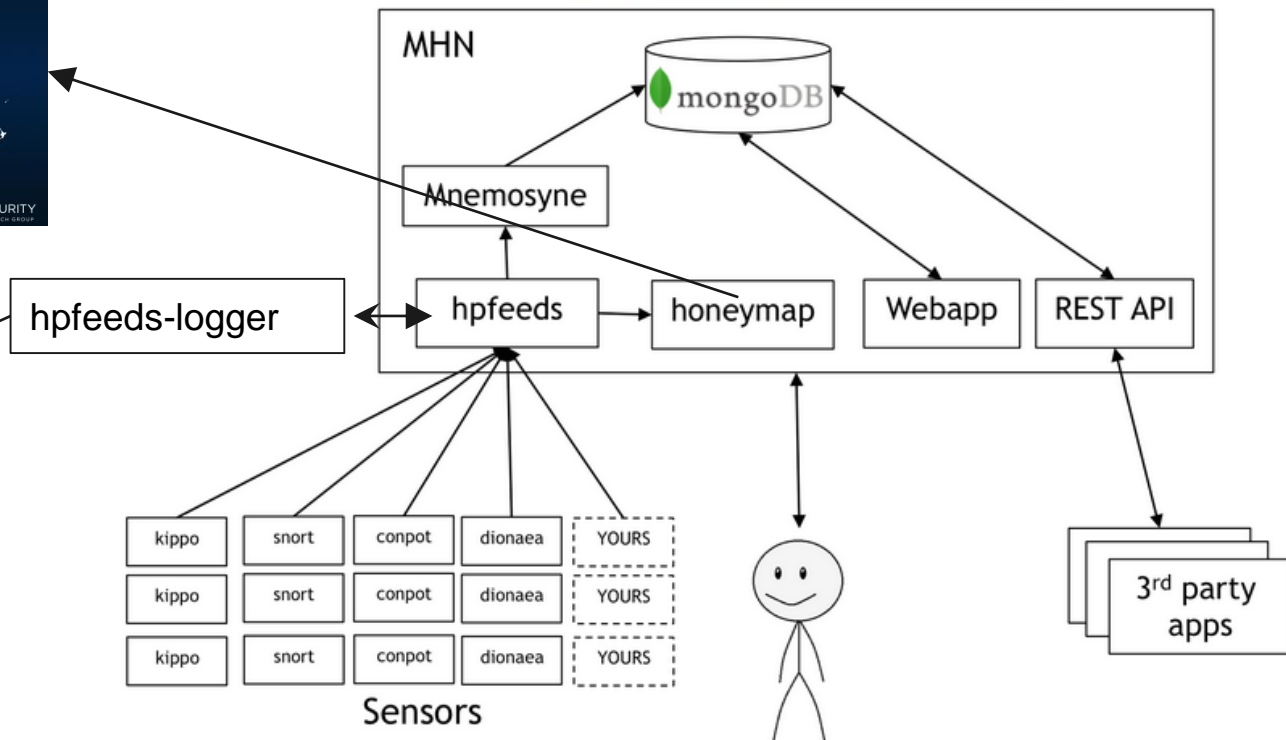
# Conclusion

Sujet riche

Nouvelles technologie, exploration de protocols industriel

# Questions ?

# Repos

https://github.com/lightoyou/HoneyPot_CDSI

https://github.com/lightoyou/HoneyPot_CDSI_pymodbus