



# Ça fait quoi si j'appuie là ?

Retour d'expérience de tests d'intrusion  
sur systèmes industriels

Claire Vacherot @ SSTIC 2025



## Claire Vacherot

Pentester & researcher @ Orange Cyberdefense

- ▶ Tests d'intrusion, spécialité systèmes industriels
- ▶ Recherche sécurité des réseaux et systèmes industriels
- ▶ Speaker @ GreHack, Defcon, Pass The Salt, Hack.lu, et maintenant au SSTIC !



# Sommaire

## Retour d'expérience de tests d'intrusion sur systèmes industriels\*

- ▶ Présentation
- ▶ Méthodologie
- ▶ Techniques
- ▶ Observations

### \* Anonymisé

Aucun exemple ou image montré n'est issu d'un de nos audits.  
Mais tous sont inspirés de ce qu'on y a vu.



Régis sur son lieu de travail

# Systemes industriels

Composants matériels et logiciels permettant de contrôler des **procédés physiques et mécaniques**

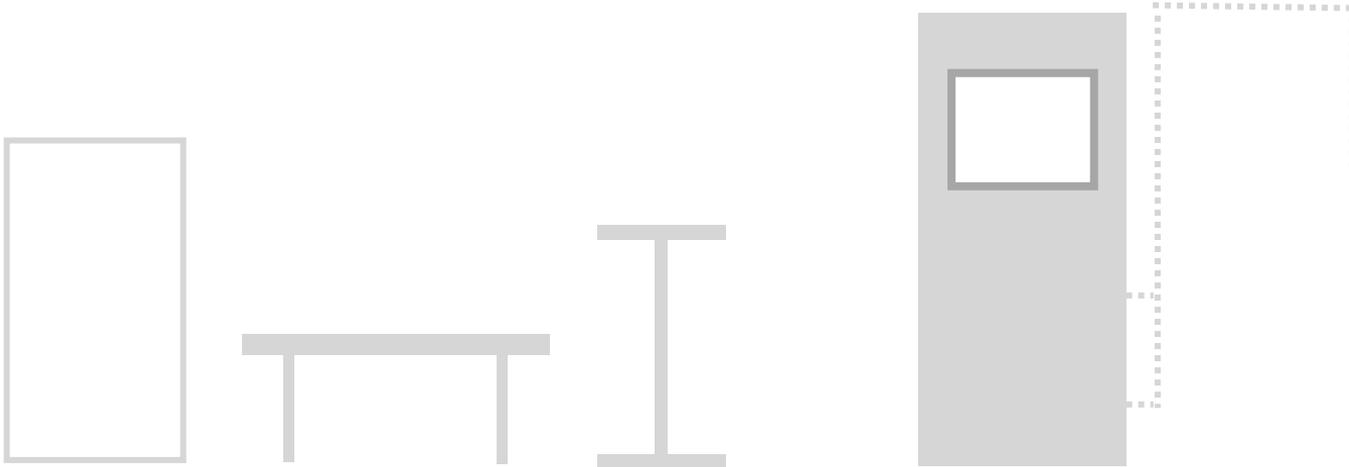


# Exemples de systèmes industriels



# Composants d'un système industriel (OT)

- ▶ Diversité des environnements = pas de schéma « type » possible
- ▶ OT = Operational Technology

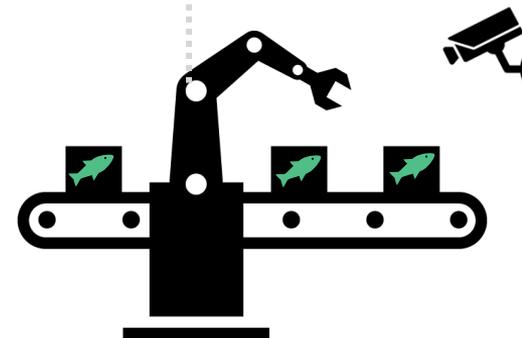
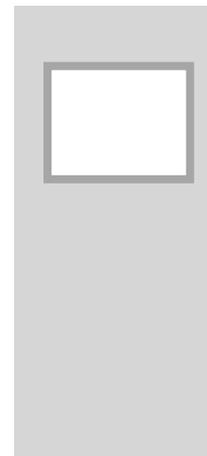
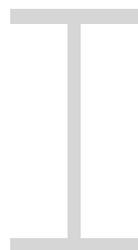


# Composants d'un système industriel (OT)

## Capteurs et actionneurs



Capteurs  
Actionneurs

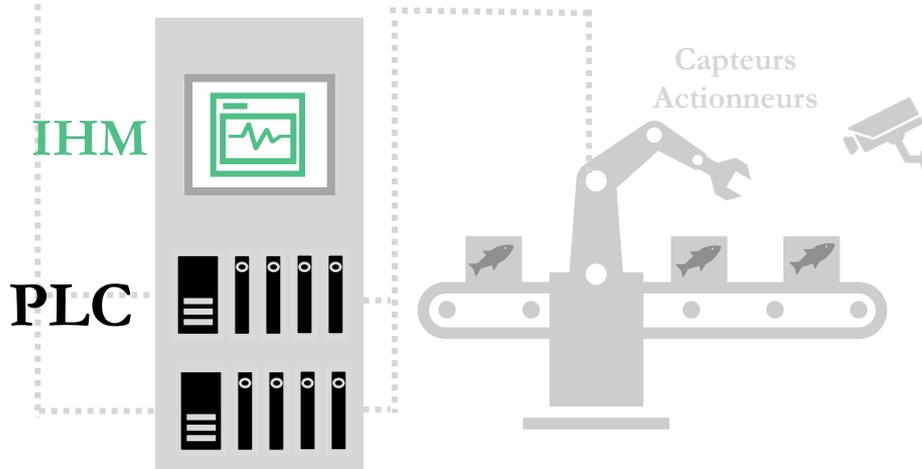


Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Composants d'un système industriel (OT)

## Automates programmables (PLC)

IHM, RTU et autres...

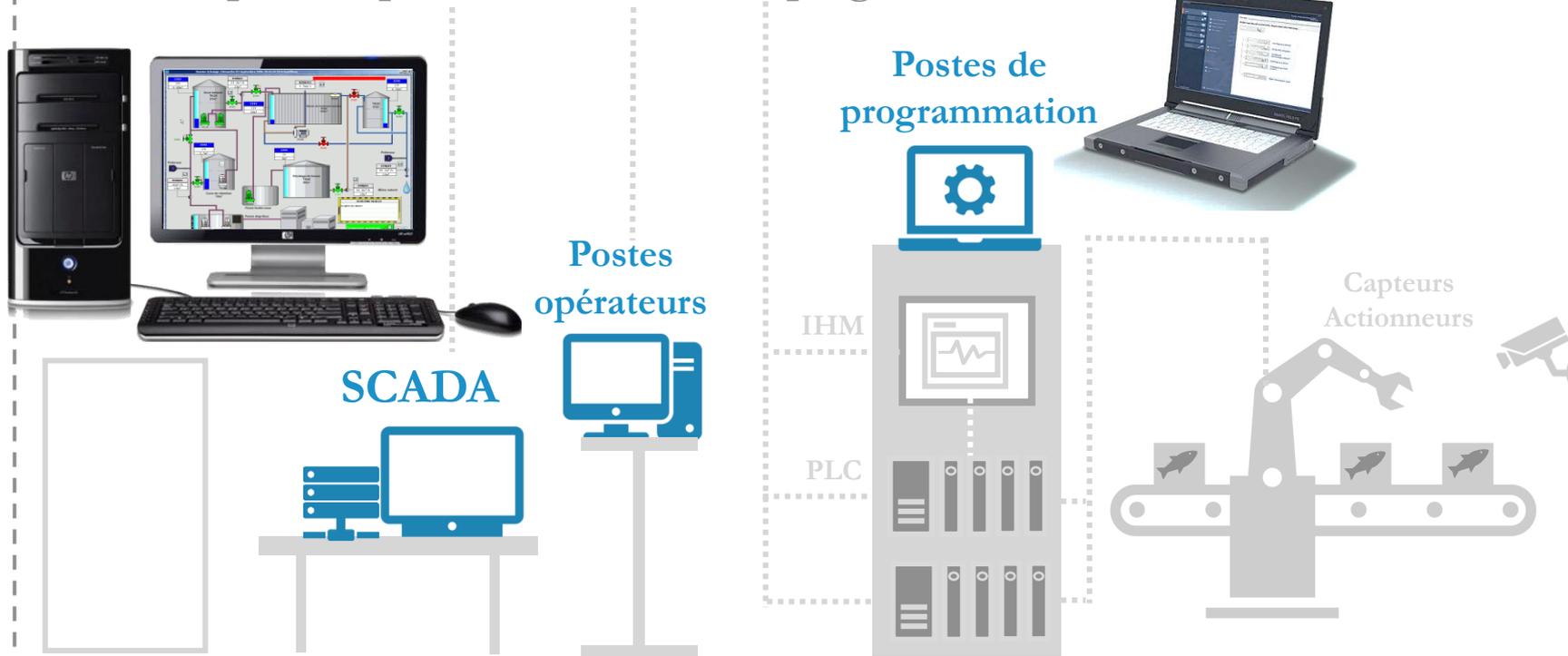


Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Composants d'un système industriel (OT)

## Postes / serveurs informatiques

SCADA, postes opérateurs, stations de programmation, etc.



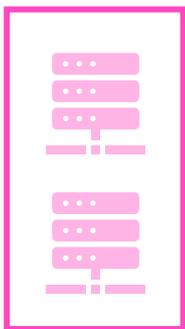
Attention: Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Composants d'un système industriel (OT)

## Serveurs

Opérations industrielles (MES, Historian, ...), gestion de parc, etc.

## Serveurs



Postes de programmation



IHM

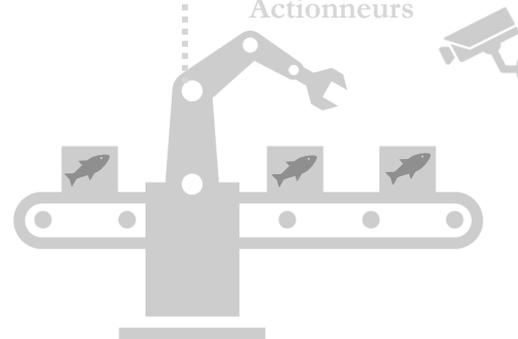


PLC



Capteurs

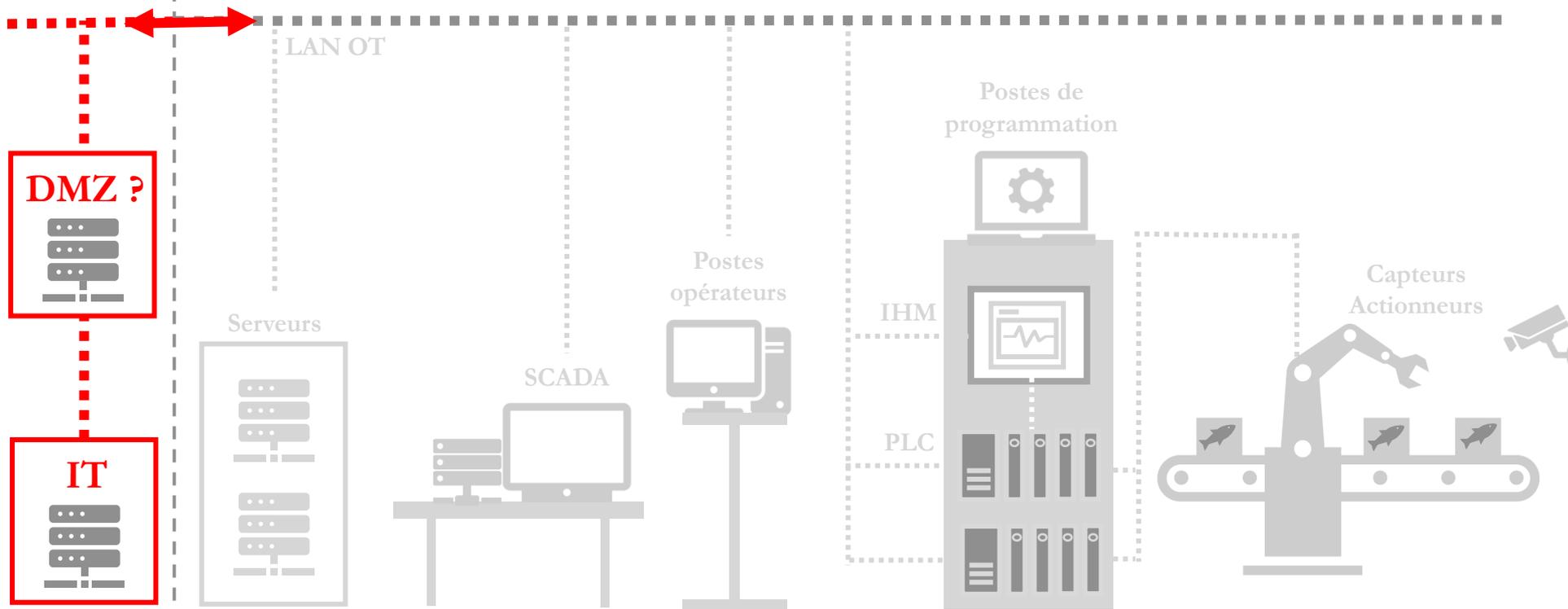
Actionneurs



Attention: Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Réseau(x) d'un système industriel (OT)

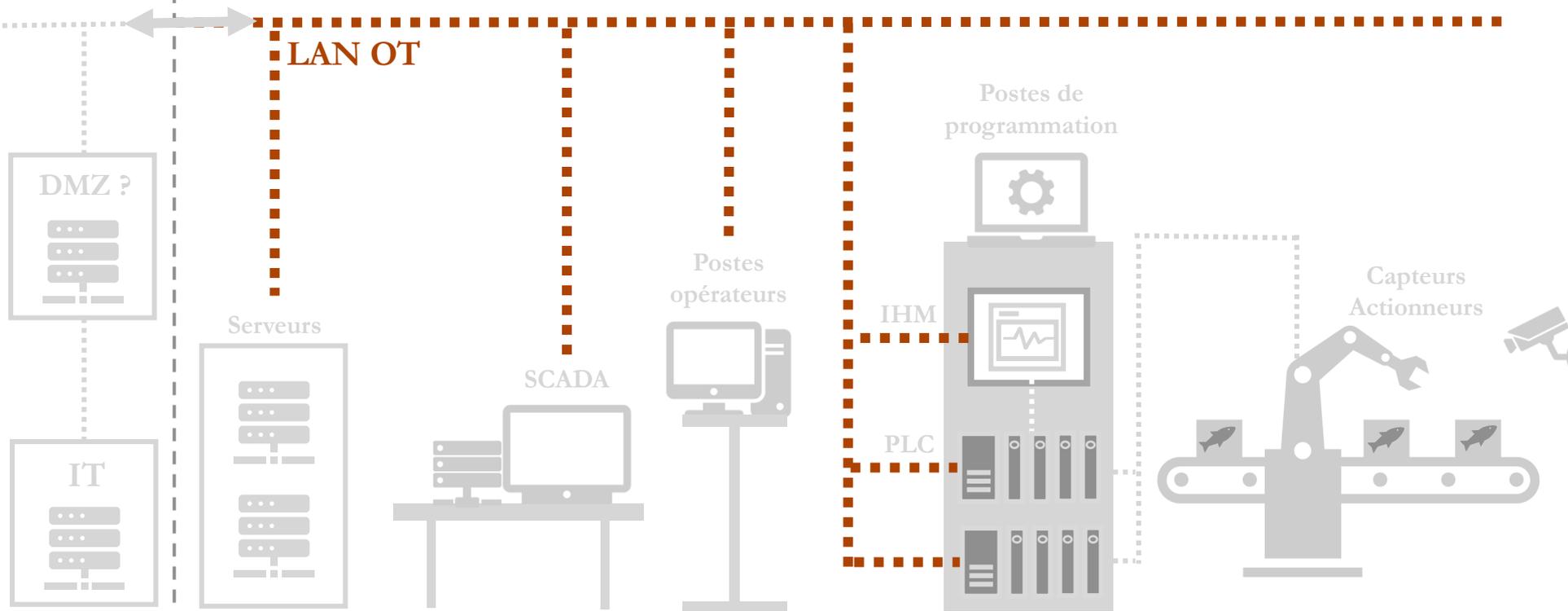
Lien IT / OT



Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Réseau(x) d'un système industriel (OT)

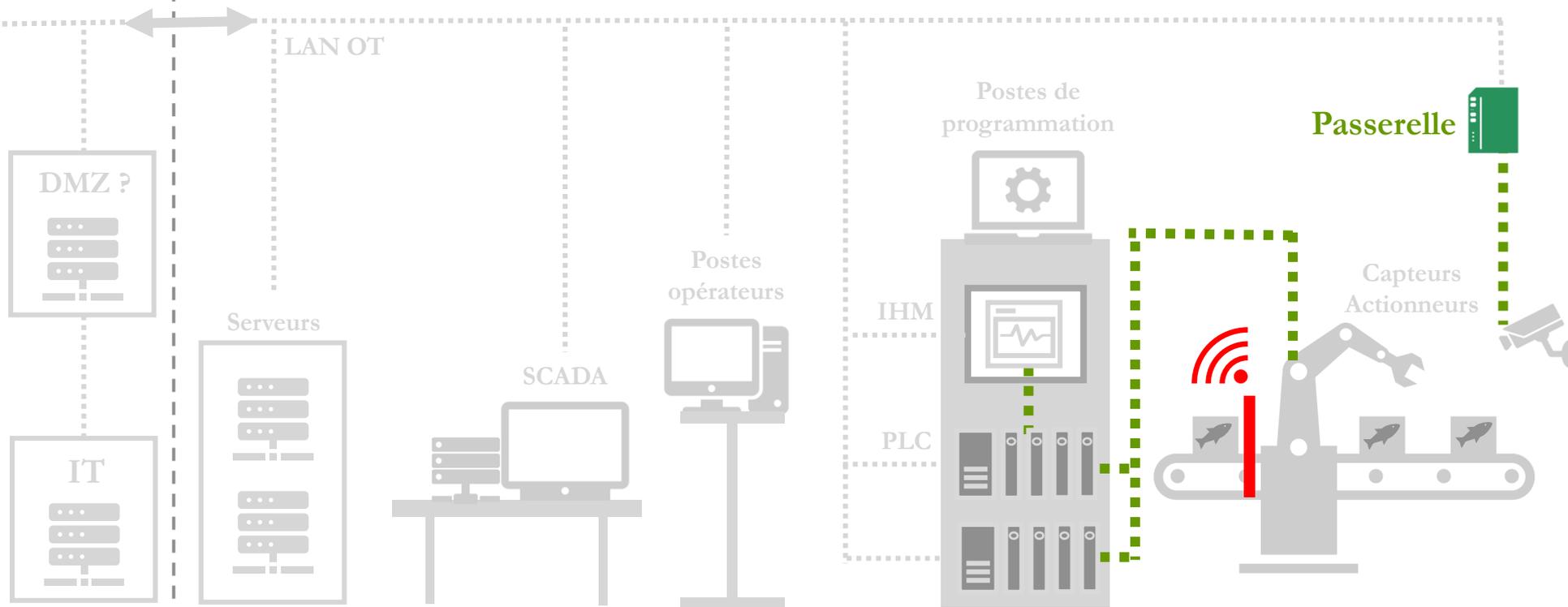
Réseau IP OT filaire et/ou sans fil



Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

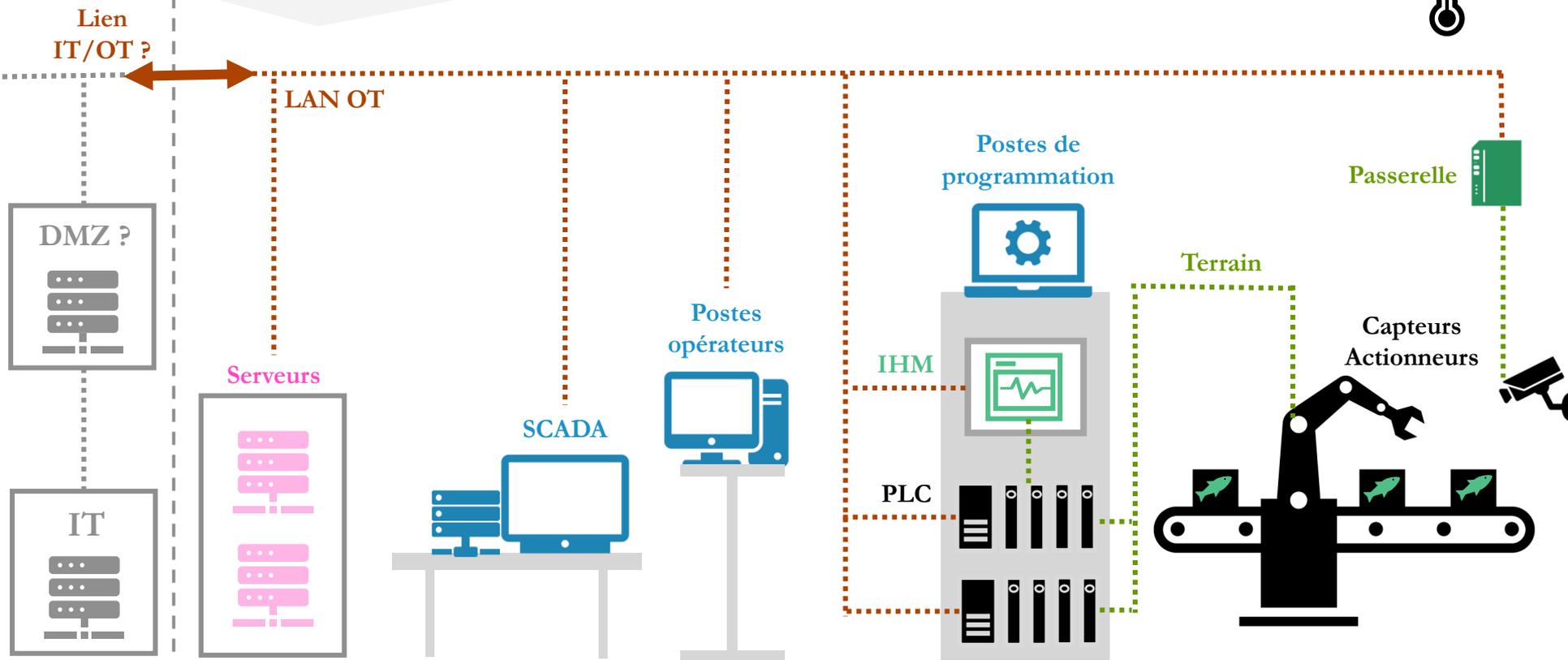
# Réseau(x) d'un système industriel (OT)

Réseau terrain OT filaire et/ou sans fil et IoT



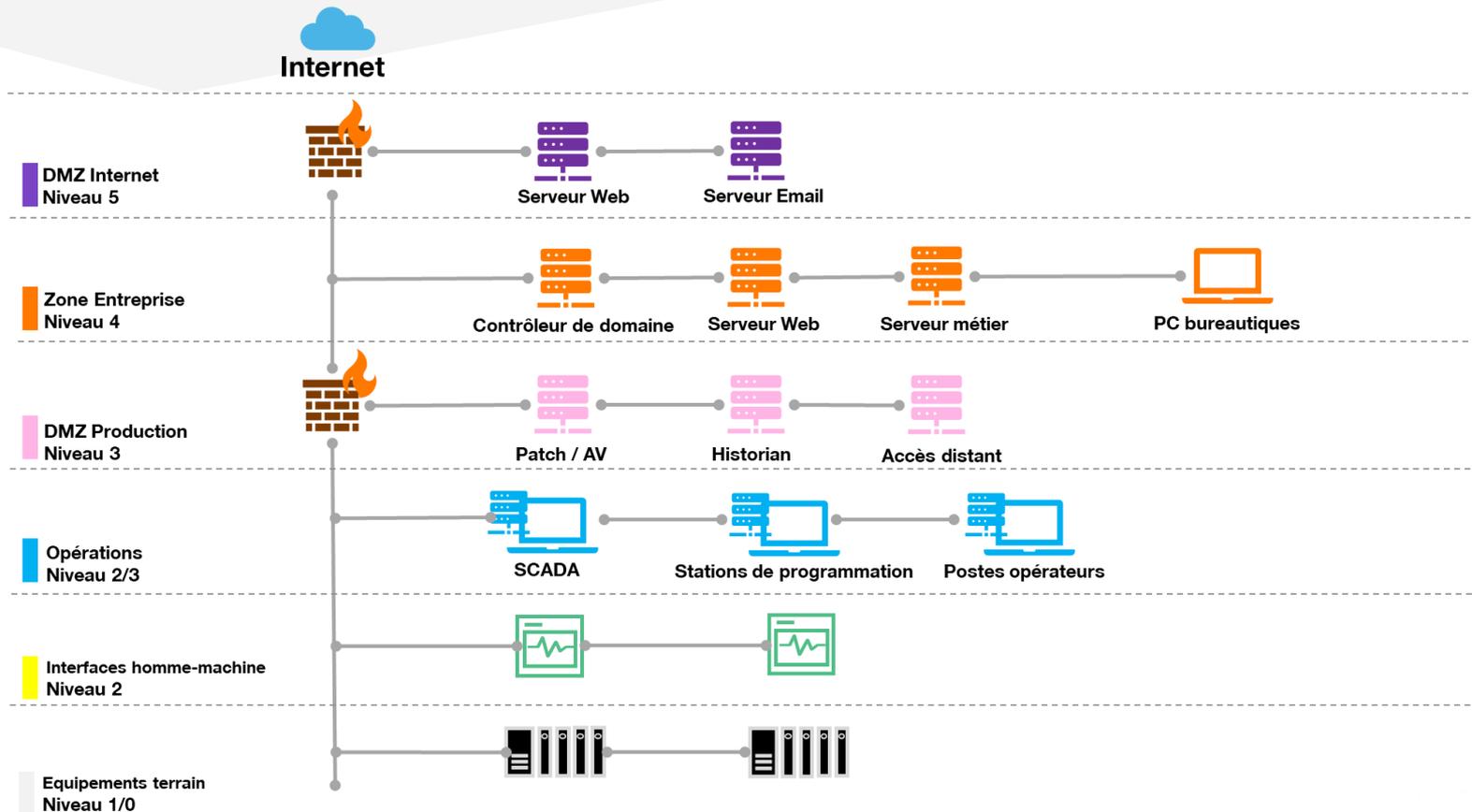
Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Composants d'un système industriel



Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Purdue model



# Cybersécurité



# Peut-on tout faire péter ?

Normalement non.

- ▶ Safety Instrumented Systems (SIS)
- ▶ Autres équipements / mesures de **sûreté**



Mais...

- ▶ Mesures pas infailibles
- ▶ D'autres moyens de causer des dégâts



# Menaces ciblant les systèmes industriels

Large panel d'enjeux techniques et métiers = pas de généralisation possible



Exemple

Chaîne en « flux tendu »



Exemple

Solutions chimiques



Exemple

Lots pharmaceutiques



Exemple

Recherche industrielle



- ▶ Dégâts matériels et humains
- ▶ Retrait d'homologation
- ▶ Interdiction de mise sur le marché
- ▶ Pertes financières

# Conditions de test

WARNING  
PROD

- ▶ Tests d'intrusion industriels normalement en environnement de test

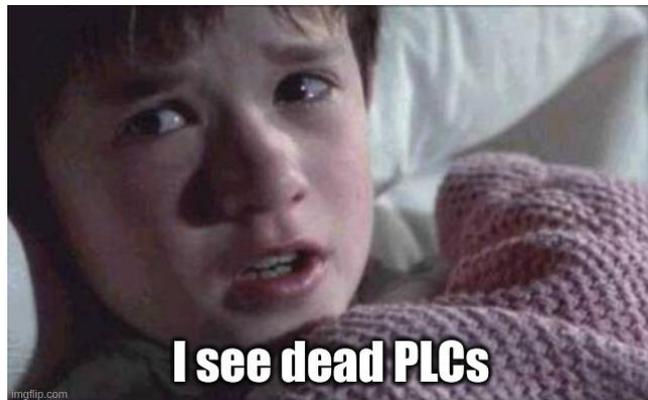
Souvent inexistant ou pas représentatif

- ▶ La majorité de nos tests sont en **production\***

Malgré nous...

- ▶ **Impact sur la portée et la méthodologie**

On ne peut pas faire « comme d'habitude »



\* « La production » : Environnement en fonctionnement

# Tests d'intrusion OT

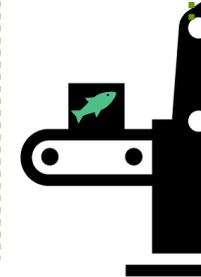
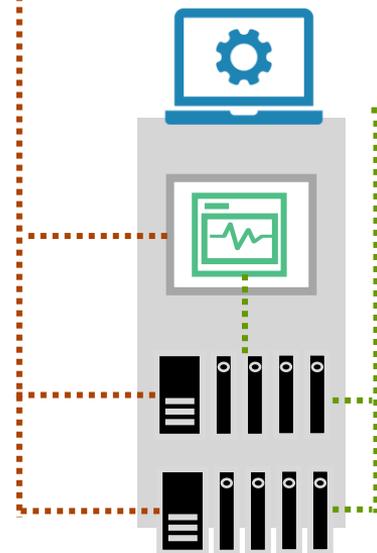
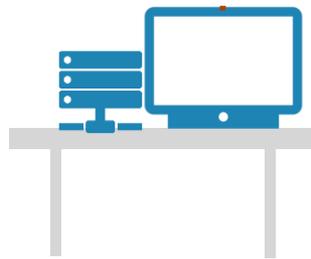
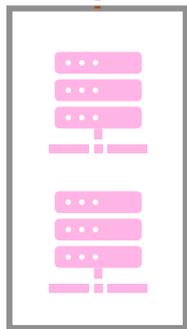
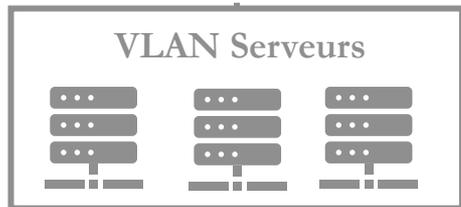
(en production)



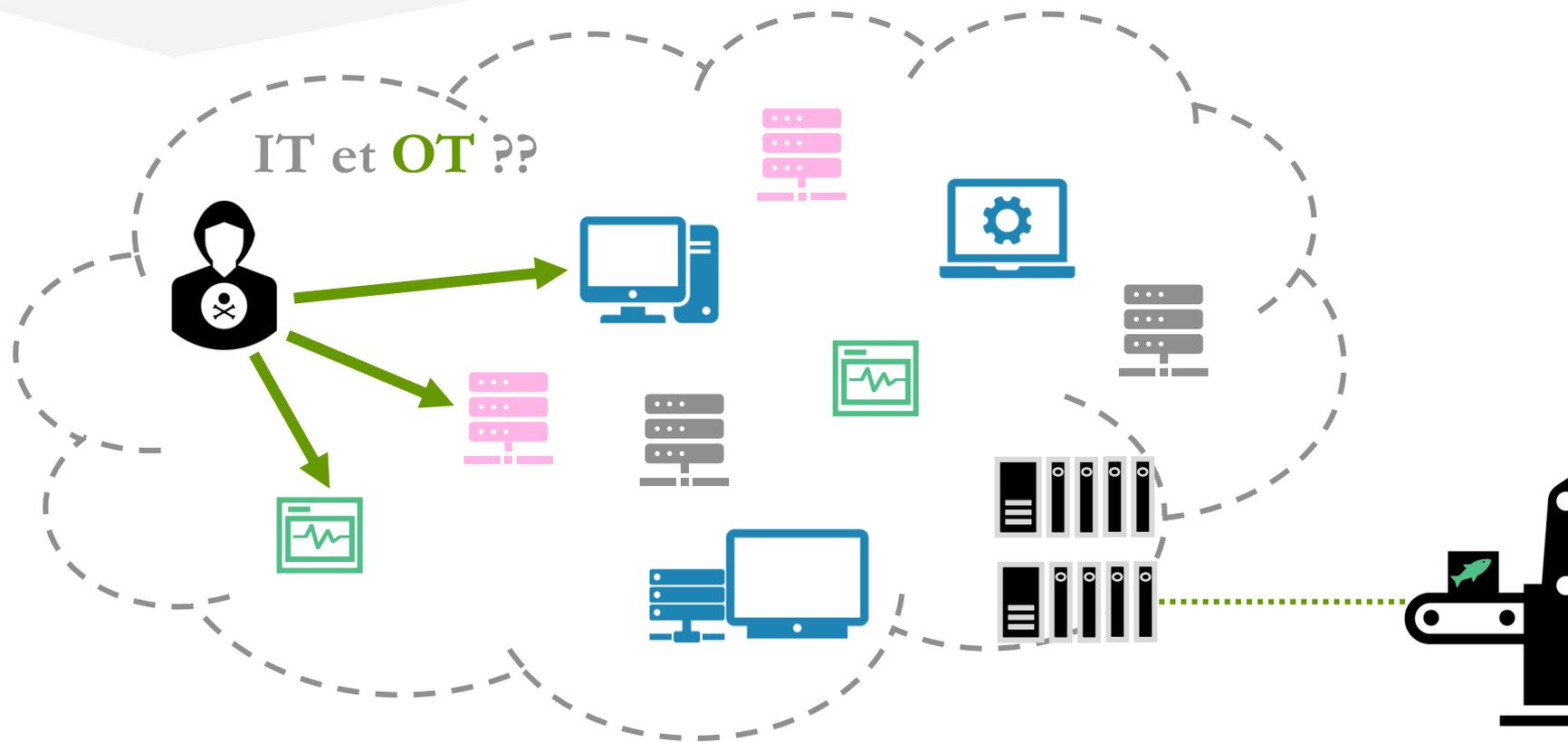
# Passage de l'IT à l'OT

IT OT

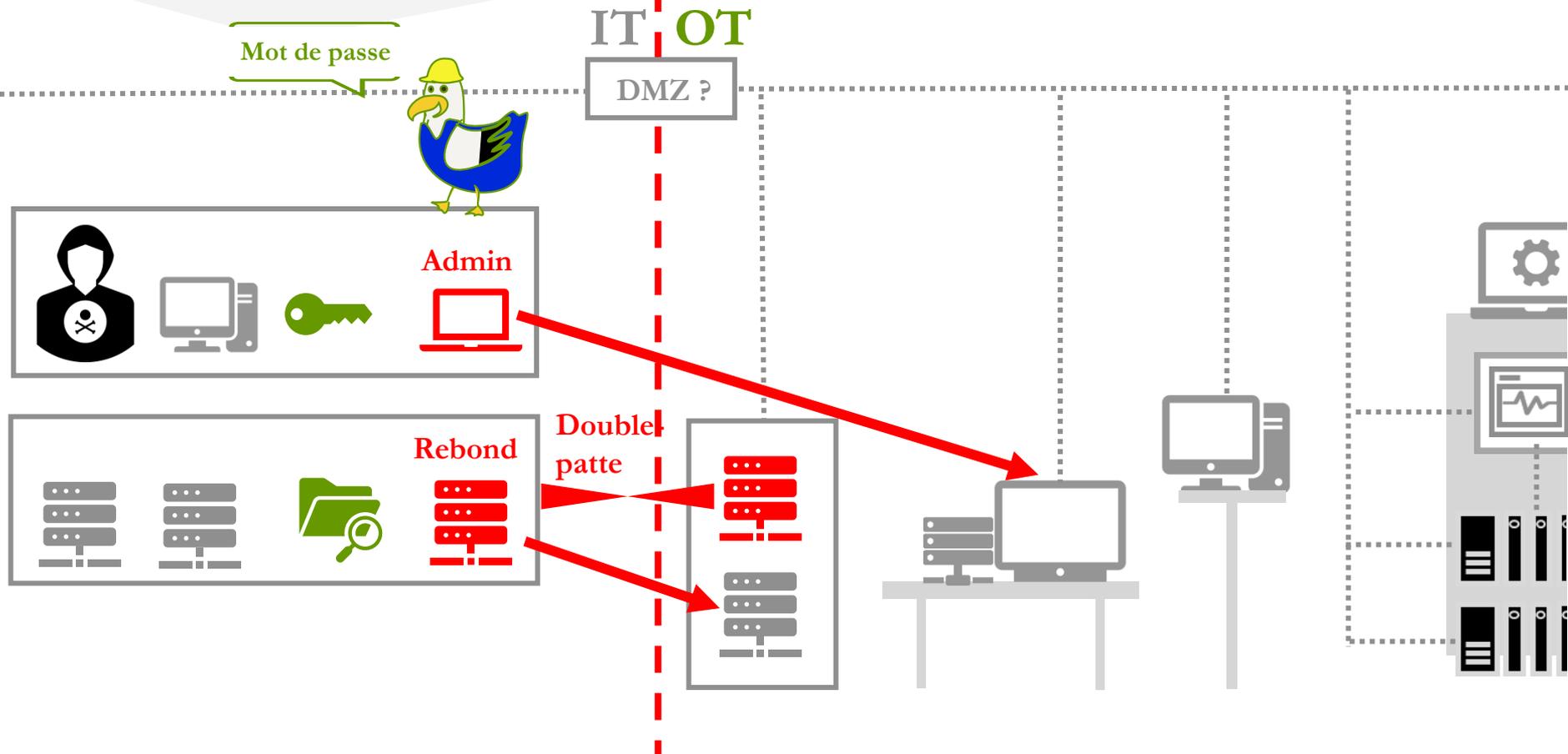
DMZ ?



# Et parfois...



# Points de passage fréquents



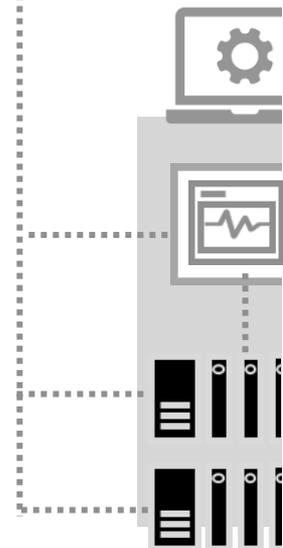
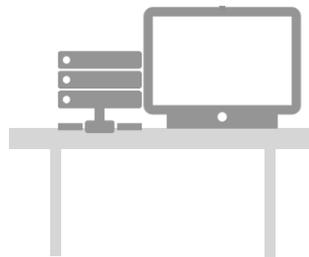
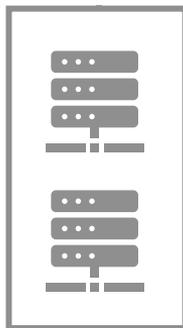
# « Pivot »

IT OT

DMZ ?



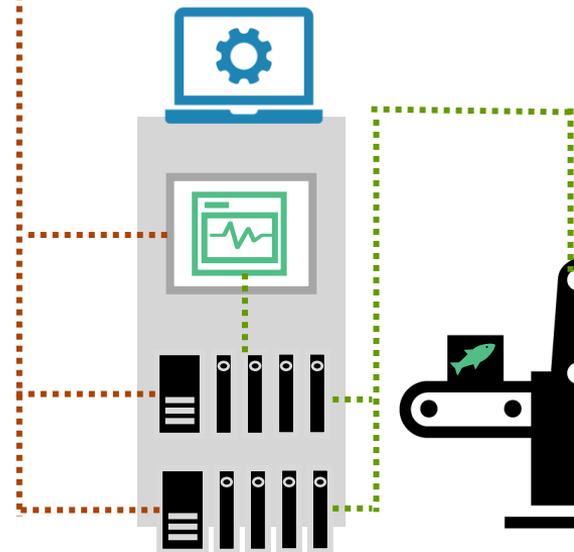
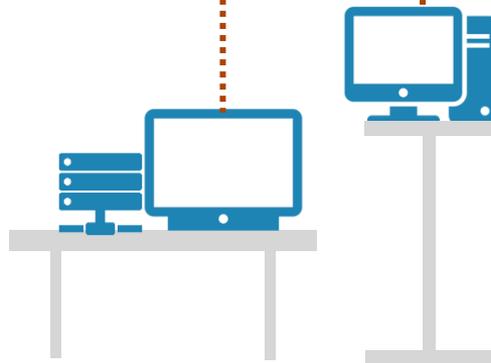
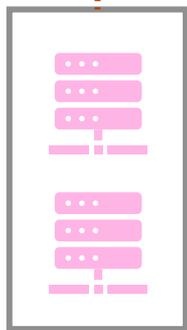
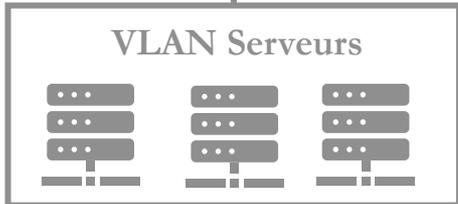
Tunnel SSH / outils de pivot



# Vous êtes ici

IT OT

DMZ ?



# Déroulement des tests d'intrusion

- ▶ Phase de découverte
- ▶ Recherche de vulnérabilités
- ▶ Synthèse et scénarios
- ▶ Restitution des résultats



**Vous êtes ici en production**

**WARNING  
PROD**

IT OT

DMZ ?

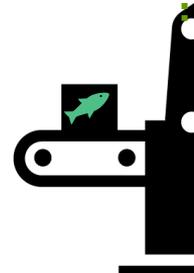
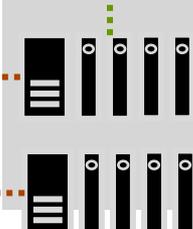
VLAN utilisateurs



VLAN Serveurs



DIRECTED BY  
MICHAEL BAY



# Précautions

WARNING  
PROD

- ▶ **Obtention d'informations**

Schéma d'architecture, plan d'adressage, etc.

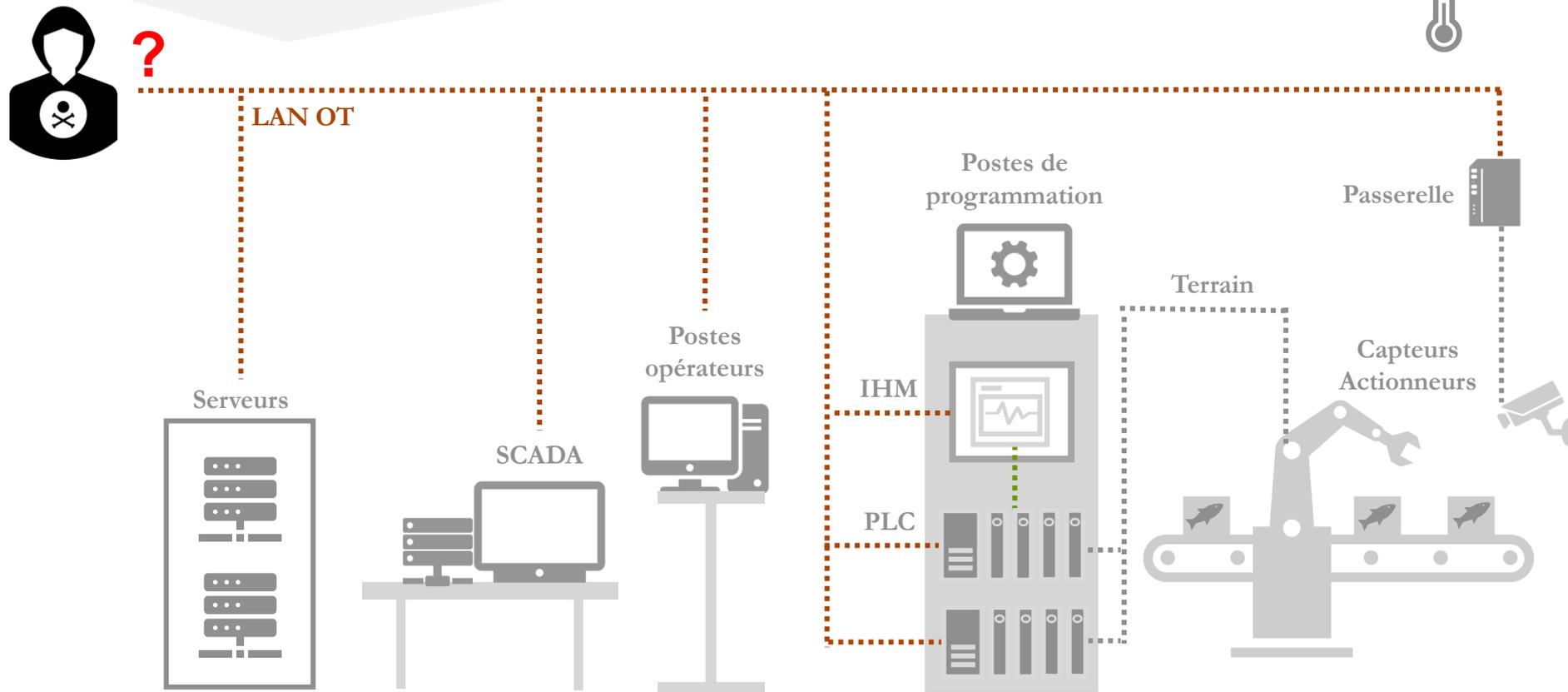
- ▶ **Exclusion des composants critiques**

Genre ça →



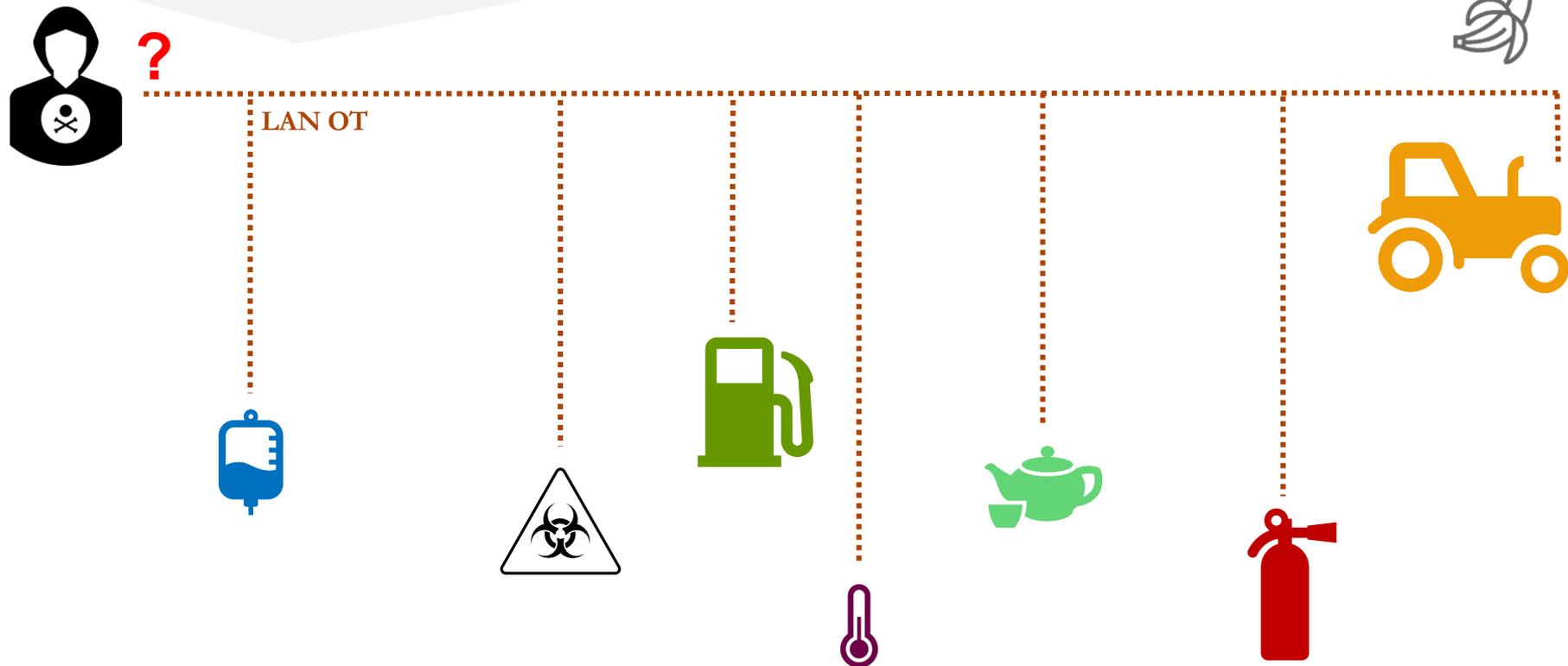
Ce sont des exemples, la robustesse de ces modèles n'est pas connue (sauf Windows 98)

# Phase de découverte



Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Phase de découverte



Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Phase de découverte

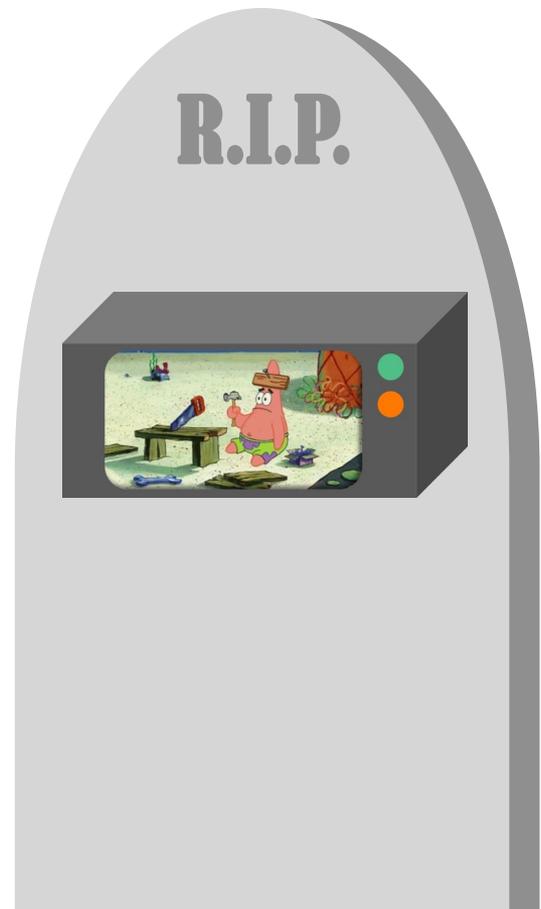
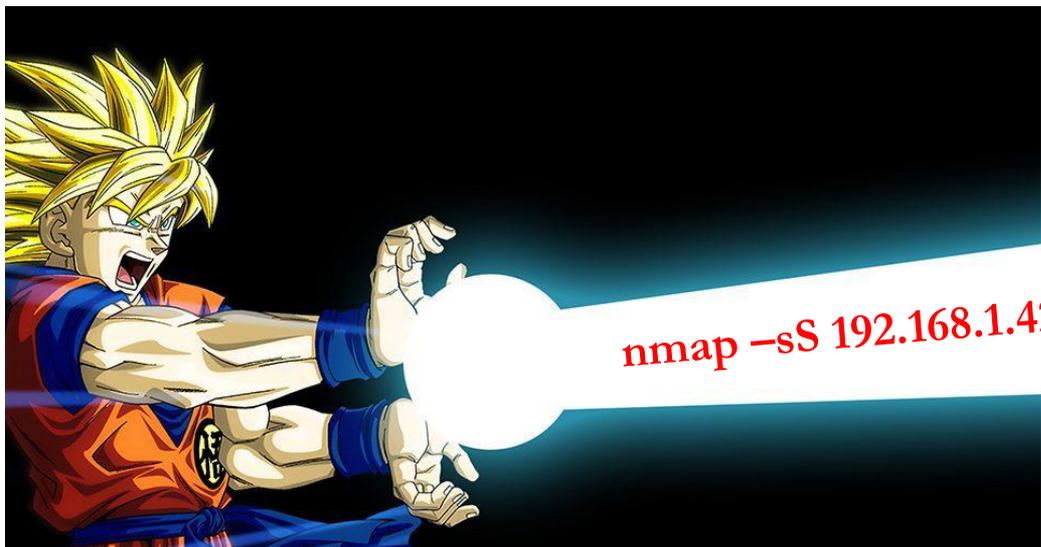
- ▶ Cycle de vie des composants
- ▶ Equipements très spécialisés
- ▶ Fonctionnement et sûreté des procédés
- ▶ Validation et homologation



WARNING  
PROD

# Scans réseaux ?

WARNING  
PROD



# Techniques de découverte « légère »

## ► Plein de choses à faire

Ex : interroger les équipements réseau

Time	Source	Destination	Protocol	Length	Info
1959	217.305942997	HMSIndus_80:03:80	LLDP_Multicast	76	MA/00:30:11:80:03:80 IN/Port 1 120
2011	219.915068594	HMSIndus_80:03:84	LLDP_Multicast	129	MA/00:30:11:80:03:83 LA/port-001 20 RTClas
2014	220.984926786	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2890, Delay=
2015	221.184883290	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2891, Delay=
2016	221.384934516	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2892, Delay=
2017	221.584909907	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2893, Delay=
2018	221.784920195	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2894, Delay=
2029	224.964983142	HMSIndus_80:03:84	LLDP_Multicast	129	MA/00:30:11:80:03:83 LA/port-001 20 RTClas
2049	229.984907933	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2895, Delay=
2050	230.015014817	HMSIndus_80:03:84	LLDP_Multicast	129	MA/00:30:11:80:03:83 LA/port-001 20 RTClas
2051	230.184909549	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2896, Delay=
2052	230.384907712	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2897, Delay=
2054	230.584913965	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2898, Delay=
2055	230.784905371	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2899, Delay=
2071	235.064971646	HMSIndus_80:03:84	LLDP_Multicast	129	MA/00:30:11:80:03:83 LA/port-001 20 RTClas
2162	238.984965094	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2900, Delay=
2169	239.184871138	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2901, Delay=
2181	239.384913836	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2902, Delay=
2183	239.584898417	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2903, Delay=
2186	239.784906715	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2904, Delay=
2216	240.114777013	HMSIndus_80:03:84	LLDP_Multicast	129	MA/00:30:11:80:03:83 LA/port-001 20 RTClas
3592	243.852635684	192.168.1.19	HICP	54	Request message, Command: Module scan
3593	243.852638319	192.168.1.19	HICP	54	Request message, Command: Module scan
4570	245.164792987	HMSIndus_80:03:84	LLDP_Multicast	129	MA/00:30:11:80:03:83 LA/port-001 20 RTClas
6127	247.306928223	HMSIndus_80:03:80	LLDP_Multicast	76	MA/00:30:11:80:03:80 IN/Port 1 120
6281	247.984675464	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2905, Delay=
6332	248.184677785	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2906, Delay=
6409	248.384672335	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2907, Delay=
6457	248.584671261	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2908, Delay=
6496	248.784874152	HMSIndus_80:03:84	LLDP_Multicast	60	DelayReq , Seq=2909, Delay=

## ► Broadcast

Choix historique, architecture réseau

Dissecteurs Wireshark <3

## ► Multicast

Supporté par plein de protocoles OT

# Multicast avec KNXnet/IP

WARNING  
PROD

Multicast 224.0.12.23

Search  
Request



KNX gateway floor 1  
192.168.1.2  
0.1.1



KNX gateway elevator  
192.168.1.11  
0.0.10



IP-Interface Secure  
192.168.1.23  
15.15.255



```
Internet Protocol Version 4, Src: 192.168.1.23, Dst: 192.168.1.17
User Datagram Protocol, Src Port: 3671, Dst Port: 60894
KNX/IP Search Response, Control 0 192.168.1.23:3671, 15.15.255 "IP-Interface Secure"
- KNX/IP Header: Search Response
  Header Length: 6 bytes
  Protocol Version: 1.0
  - Service Identifier: Search Response (0x0202)
    Service Family: Core (0x02)
    Service Type: Search Response (0x0202)
    Total Length: 78 bytes
  - HPAI Control Endpoint: 192.168.1.23:3671 UDP
    Structure Length: 8 bytes
    Host Protocol: IPv4 UDP (0x01)
    IP Address: 192.168.1.23
    Port Number: 3671
  - DIB DevInfo: 15.15.255 "IP-Interface Secure"
  - DIB SuppSvc: Tunneling, Remote Diag And Config
```

# Protocoles réseaux industriels

- ▶ Surveiller, contrôler, configurer
- ▶ Il y en a **BEAUCOUP**  
Selon constructeurs, secteurs, usage...
- ▶ Liste incomplète ici :



[github.com/Orange-Cyberdefense/  
awesome-industrial-protocols](https://github.com/Orange-Cyberdefense/awesome-industrial-protocols)



# Peut-on utiliser ces protocoles en pentest ?



Risques d'effets de bord élevés

Usage maîtrisé ou **hors production** seulement



L'un des composants principaux des systèmes industriels

Ce serait dommage de passer à côté...

# Quelques techniques

## ► Outils et modules existants

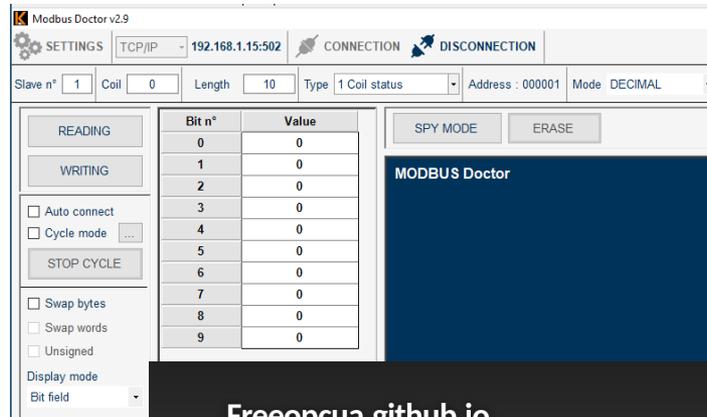
Open source, scripts nmap internes et externes

## ► Développement de scripts

Layers Scapy pour certains protocoles



## ► A valider en environnement de test



[Freeopcu.github.io](https://github.com/FreeOpcUa)

FreeOpcUa: Open Source Rust and Python OPC-UA Client and Server  
Libraries and Tools



FreeOpcUa is a project created to support the implementation and maintenance of open-source  
OPC-UA stacks and associated tools.

# Exemple : Découverte ciblée Ethernet/IP

- ▶ Nmap : enip-discover
- ▶ Scapy : enipTCP.py, requête List Identity



```
s = socket()
s.connect((argv[1], 44818))
ss = StreamSocket(s, Raw)
# Creation de la requete
pkt = ENIPTCP()
pkt.commandId = 0x63
# Envoi de la requete, reception de la reponse
resp = ss.sr1(pkt)
resp = ENIPTCP(raw(resp))
resp.show2()
```



# Exemple : Découverte ciblée Ethernet/IP

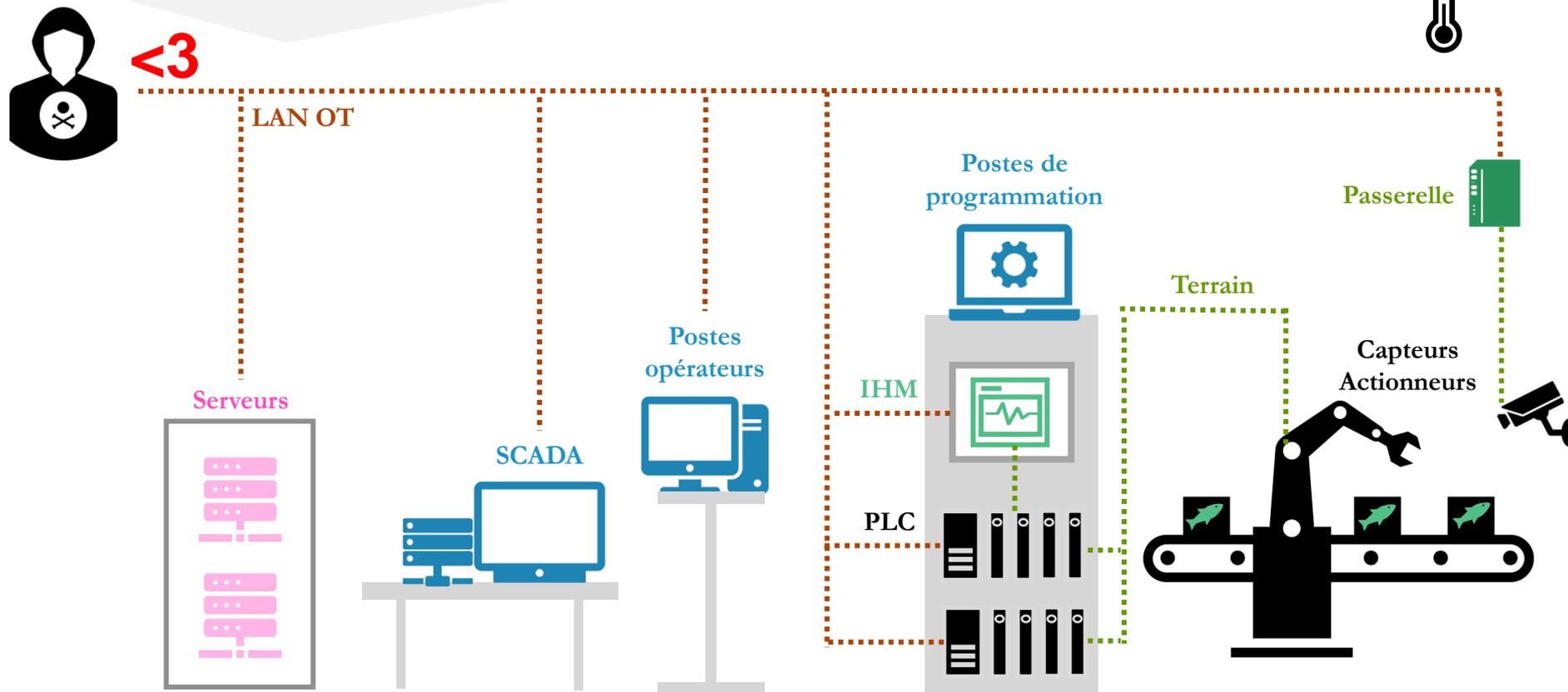


Ok merci

```
SSSTIC DEMO python enip_discovery.py 192.168.1.241
Begin emission:
Finished sending 1 packets.
*
Received 1 packets, got 1 answers, remaining 0 packets
###[ ENIPTCP ]###
  commandId = ListIdentity
  length    = 59
  session   = 0x0
  status    = success
  senderContext= 0
  options   = 0
###[ ENIPLISTIDENTITY ]###
  itemCount = 256
  \items     \
  |###[ ENIPLISTIDENTITYREPLYITEM ]###
  | itemTypeCode= CIP Identity
  | itemLength= 53
  | protocolVersion= 1
  | sinFamily = 2
  | sinPort   = 44818
  | sinAddress= 192.168.1.241
  | sinZero   = 0
  | vendorId  = 90
  | deviceType= Communications Adapter
  | productCode= 93
  | revisionMajor= 1
  | revisionMinor= 8
  | status    = 48
  | serialNumber= 0xa06f262f
  | productNameLength= 19
  | productName= 'Anybus Communicator'
  | state     = 255
```



# Après la phase de découverte...



Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Recherche de vulnérabilités

- ▶ Ne pas rendre instable les systèmes
- ▶ Ne pas altérer le fonctionnement

Comment faire ?



Régis sur son lieu de travail

WARNING  
PROD

# Première idée

## TESTER HORS PRODUCTION



## Deuxième idée

### FAIRE ATTENTION\*

- ▶ Privilégier les cibles secondaires / moins sensibles
- ▶ Connaître les conséquences possibles de ses techniques d'exploitation

\* Et demander l'accord avec le client



# Troisième idée

WARNING  
PROD

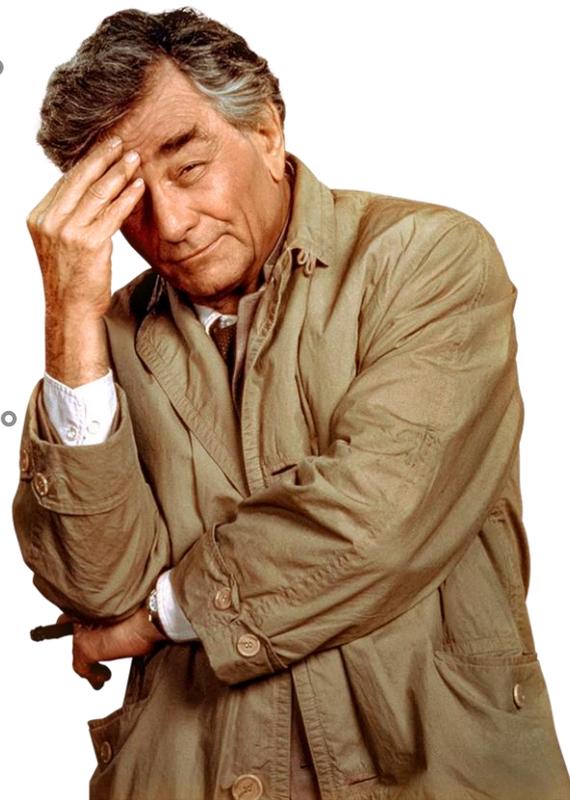
## UNE INCROYABLE CAPACITÉ DE DÉDUCTION

Un firmware de  
2009 concerné  
par 24 CVE ?

Un protocole sans  
authentification  
pour contrôler le  
procédé ?

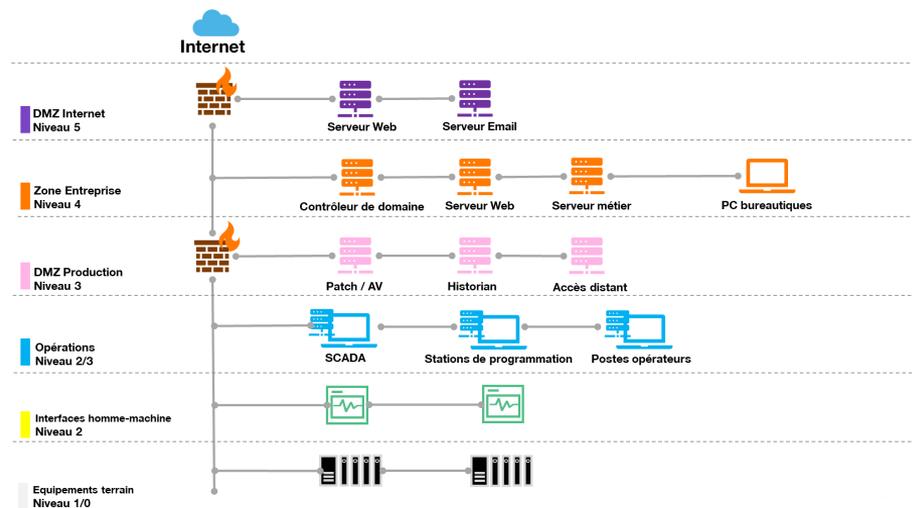
Un serveur SSH  
sans mot de  
passe pour root ?

(on peut aussi demander au client)



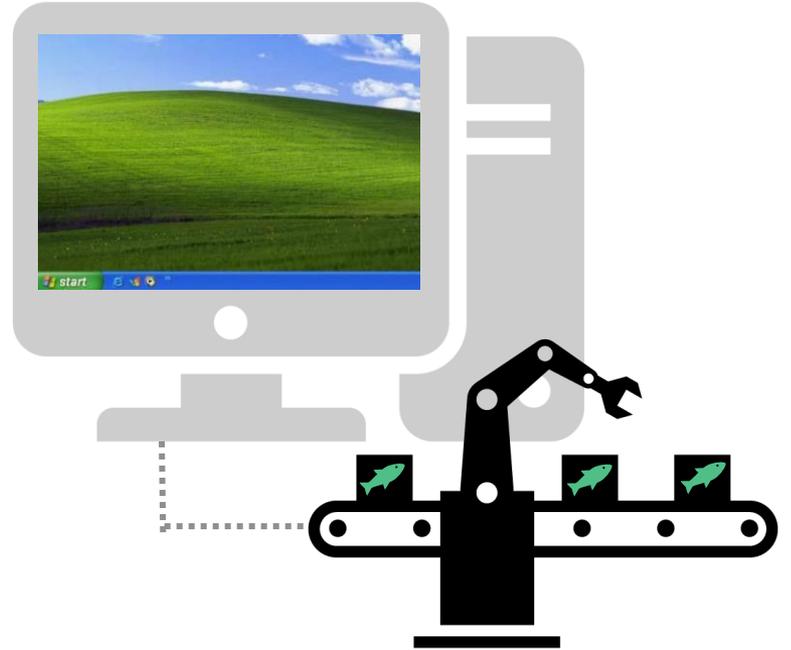
# Résultats fréquents

## ► Architecture et cloisonnement réseau



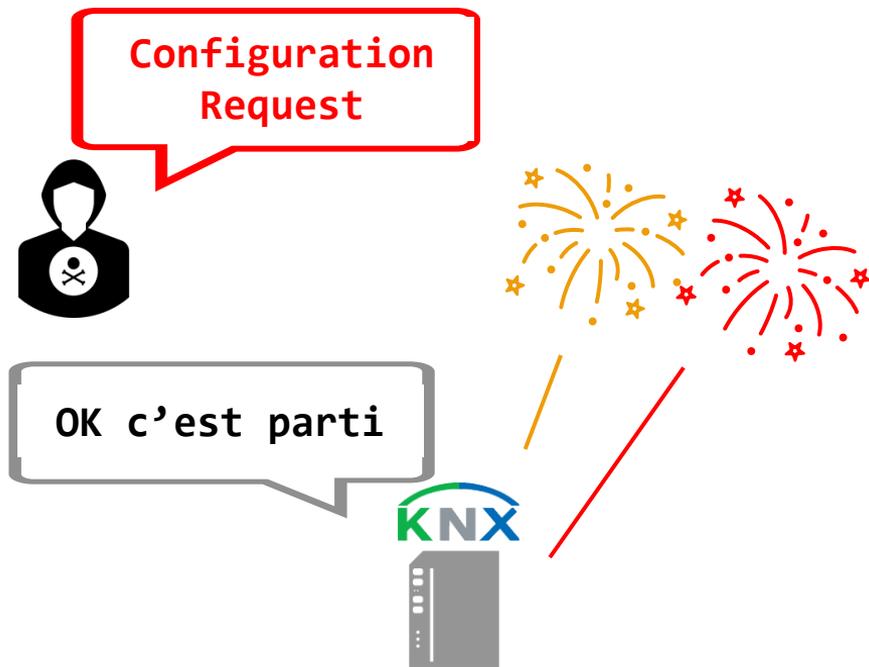
# Résultats fréquents

- ▶ Architecture et cloisonnement réseau
- ▶ Obsolescence



# Résultats fréquents

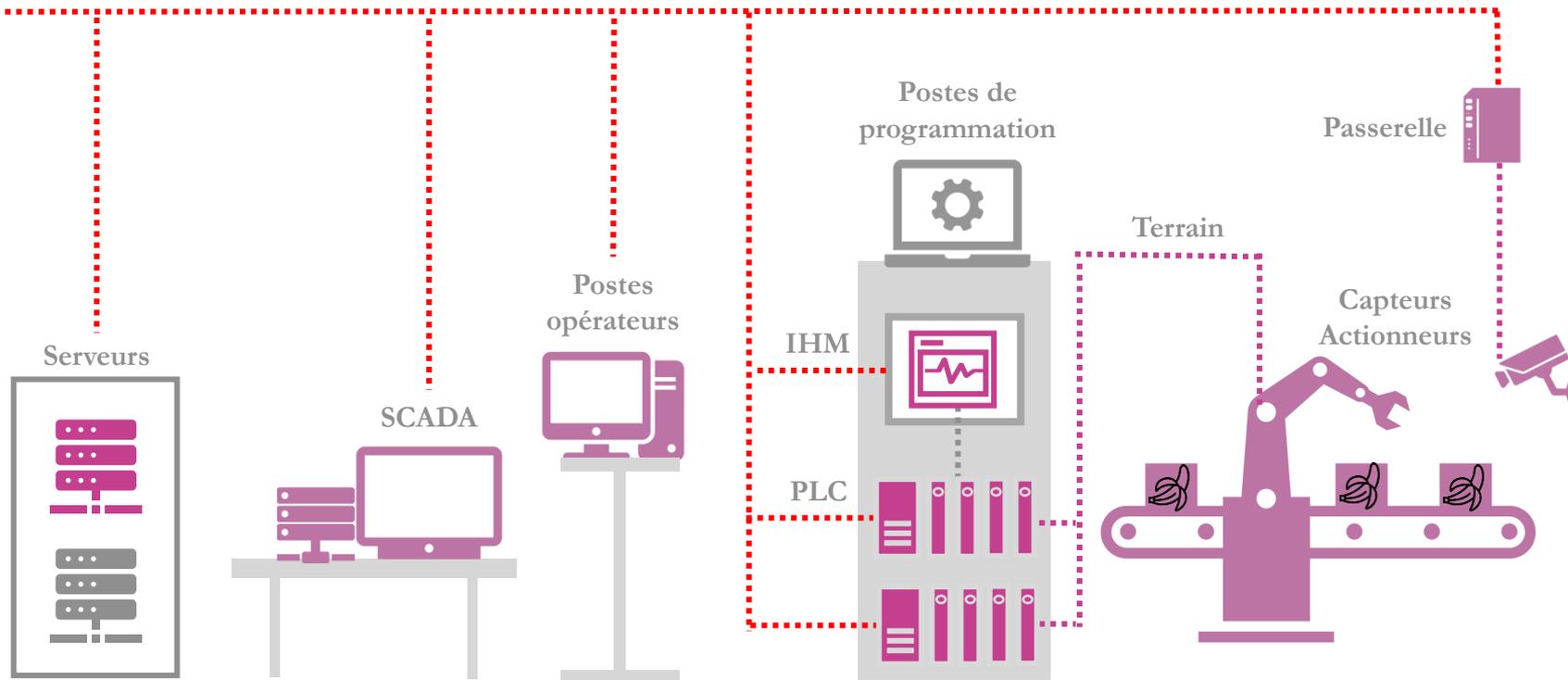
- ▶ Architecture et cloisonnement réseau
- ▶ Obsolescence
- ▶ Services non sécurisés





# Résultats fréquents

Et finalement...



Attention : Schéma simplifié, pas vraiment représentatif d'un dispositif réel

# Bilan



# Un début d'explication

- ▶ **Contraintes opérationnelles**

Cycle de vie, exigences de production, etc.

- ▶ **Sûreté de fonctionnement**

Sécurité des personnes

- ▶ **Sensibilisation des constructeurs, intégrateurs et utilisateurs finaux**

 **Bonnes pratiques de sécurité non appliquées / non applicables**



# Protéger les systèmes industriels

- ▶ **Durcissement des configurations**

A adapter selon contraintes opérationnelles / de sûreté

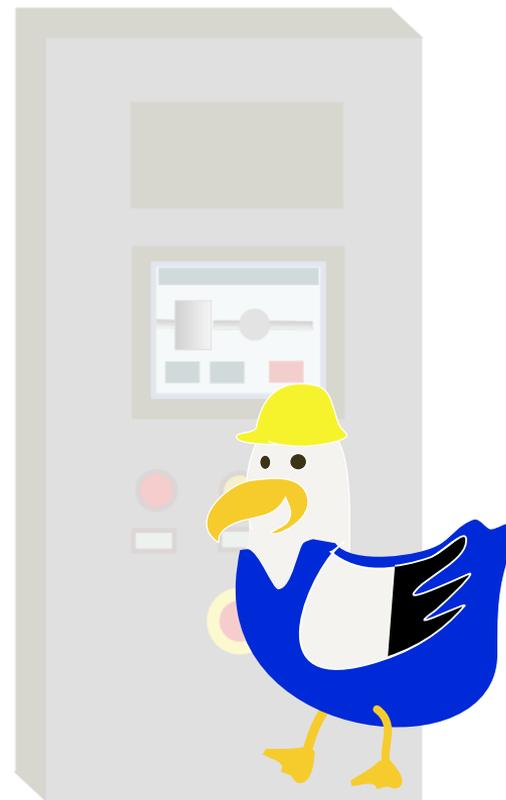
- ▶ **Solutions et procédures de sécurité**

Contrôle d'accès, gestion d'identité, détection, etc.

- ▶ **Architecture et cloisonnement réseau**

**Adapter les recommandations au contexte**

**et pas l'inverse 😊**



# Référentiels

## ANSSI

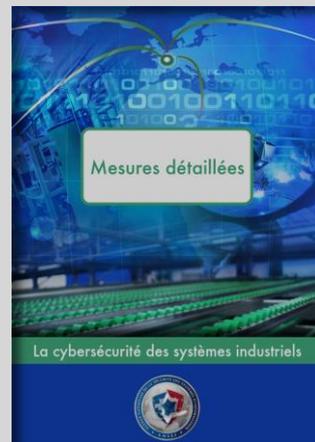
- ▶ **La cybersécurité des systèmes industriels, 2014-2025 ?**
- ▶ Recommandations relatives à l'administration sécurisée des SI, 2021
- ▶ Référentiel d'exigences de sécurité pour les prestataires d'intégration et de maintenance de systèmes industriels, 2016
- ▶ Guide pour une formation sur la cybersécurité des systèmes industriels., 2015

## NIST

- ▶ NIST SP 800-82 Rev. 3 : Guide to Operational Technology (OT) Security

...et plein d'autres !

R.I.P.



# Conclusion



Tout n'est pas perdu



Plus d'infos dans les actes



Tester un SI industriel n'est pas anodin !

Sensibilisation nécessaire

SVP faites attention



**Cyberdefense**

**Merci !**



# @non\_curat\_lex

 github/claire-lex

Ça fait quoi si j'appuie là ?  
Retour d'expérience de tests d'intrusion sur systèmes industriels