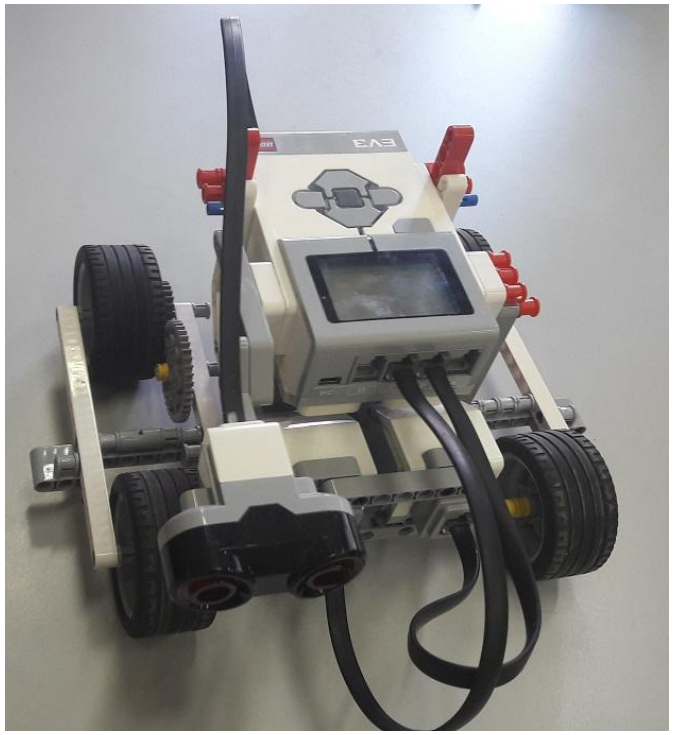


Présentation du challenge

Olivier Levillain, Grégory Blanc, Christophe Kiennert

D'après un travail de Florian Mounier, Waël Bourgou et Gabriel Jeanneau

RESSI 2019

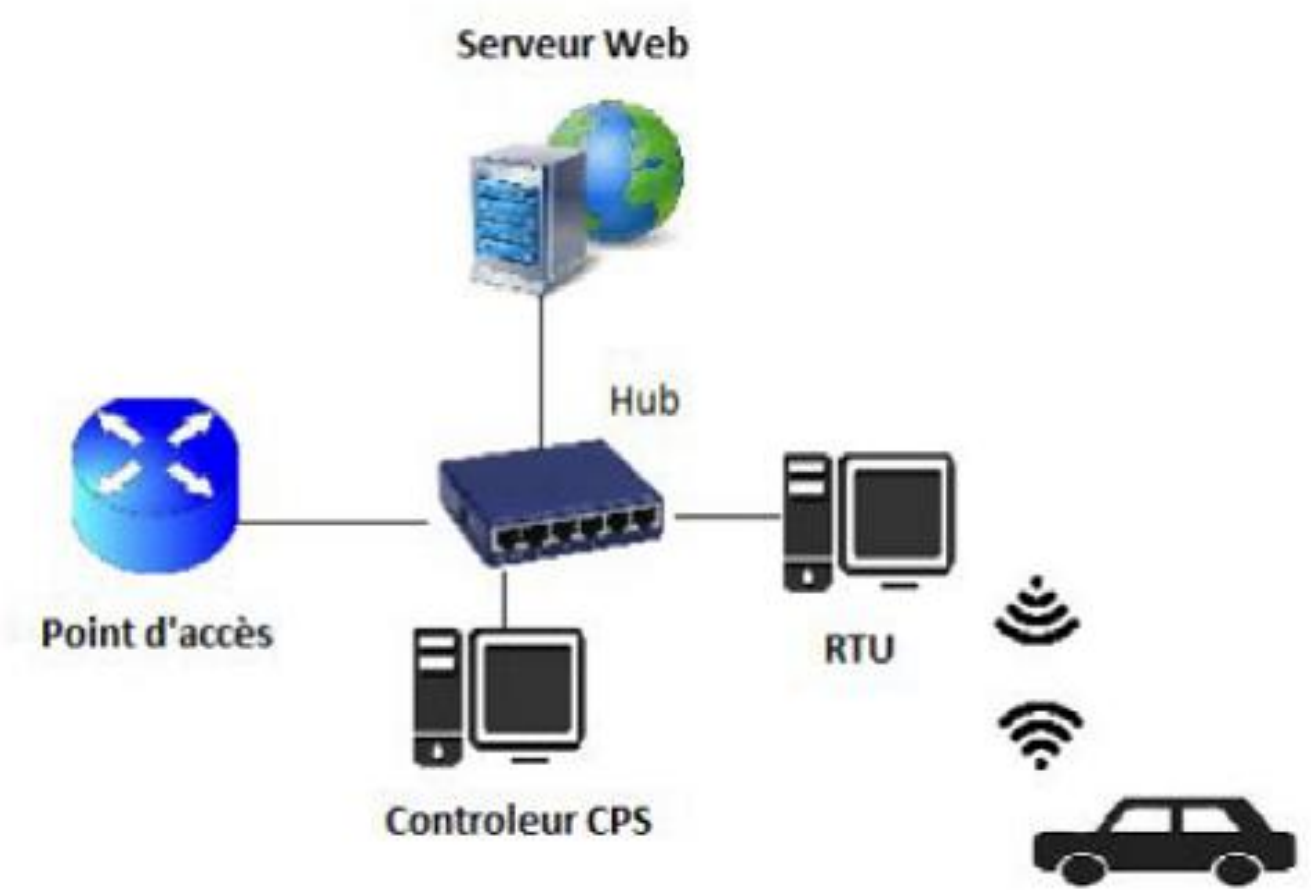
15 mai – Erquy

Scénario : durcissement d'un CPS

Giggle Car

- Une voiture autonome a été récemment mise au point par une petite start-up, conçue pour des trajets rectilignes avec détection des murs (la gestion des virages est prévue uniquement en option payante)
- Avant la mise en production des voitures, la start-up met au défi des experts en sécurité de trouver des failles sur leur système
- C'est là que votre savoir faire entre en jeu : saurez-vous ~~offrir à la start-up un audit de sécurité gratuit~~ relever ce défi ?

Architecture



Architecture

- La voiture avance en ligne droite jusqu'à rencontrer un obstacle. Elle se met alors à reculer, puis repart vers l'avant
- La voiture transmet des données au contrôleur CPS via le RTU, notamment :
  - Distance par rapport au prochain obstacle
  - Vitesse actuelle
- Le contrôleur analyse ces données et renvoie à la voiture l'un des deux ordres suivants :
  - Poursuivre dans le même sens à la même vitesse
  - Changer de sens en gardant la même vitesse
- Un serveur Web permet aux utilisateurs de laisser leurs impressions. Par ailleurs, un administrateur peut interagir avec la voiture via ce serveur.

Objectifs

- **Le challenge se déroule en deux étapes :**
  1. Recherche de vulnérabilités
  2. Durcissement du système

- **Par équipes de 2 ou de 3, vous devrez donc :**
  1. Rédiger les CVE correspondant aux vulnérabilités que vous aurez découvertes
     - Optionnellement, mettre en place des *Proofs of Concept* des vulnérabilités que vous aurez trouvées
  2. Patcher le système en corrigeant le code à disposition

Déroulement

- **Chaque équipe démarre une VM qui contient une simulation de la voiture**
  - Adresses IP des composants statiques et déjà attribuées
  - Identifiants : challenge/challenge et root/challenge
  - Chaque composant est simulé dans un LXC
- **Les CVE sont à soumettre via une interface de ticketing**
  - Attention à la qualité et la précision de la rédaction
- **Au bout d'1h30, les CVE devant faire l'objet de patchs sont publiées**
  - Les autres CVE deviennent hors périmètre
- **Modifications de code**
  - D'abord dans l'environnement de simulation
  - Puis test du code de la voiture sur la voiture réelle

Règles

1. **Faire équipe avec ses partenaires**
2. **Choisir un nom d'équipe**
3. **Respecter le périmètre imposé**
   - Aucune attaque sur le noyau de la VM ou sur l'interface de ticketing !
   - Aucune attaque physique sur la voiture !
4. **Soigner la rédaction des CVE**
5. **Tester les patchs dans l'environnement de simulation**
6. **Faire en sorte que la voiture finisse la soirée en un seul morceau**

Planning

Mer 15/5 18h – Présentation du challenge

Mer 15/5 19h – Inscriptions auprès des organisateurs

Mer 15/5 21h30 – Phase 1 : découverte des vulnérabilités

Mer 15/5 23h – Phase 2 : durcissement du système

Jeu 16/5 1h – Fin du challenge

Jeu 16/5 18h – Annonce des résultats

Le mot de la fin

Bonne chance !