Sébastien Salva
LIMOS – Université d'Auvergne

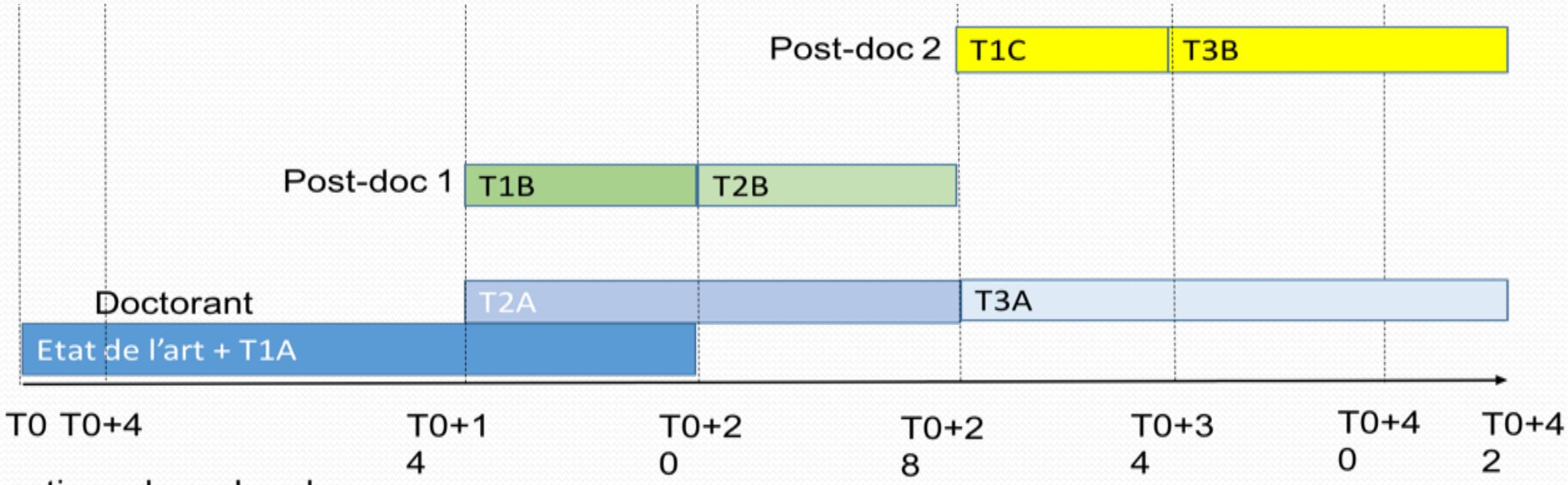
Projet VASOC

- Vers l'Audit de Sécurité des Objets Connectés

- UCA- LIMOS
  - ->software
- Université Saint-Etienne, Lab Hubert Curien
  - ->hardware

- Object-As-A-Service, Agaetis, Openium, Braincube, Elkya, Les Mowdoo

Déroulement



Allocations de recherche:
Doctorant: T0+4->T0+40 Post-doc 1 : T0+14->T0+26
Post-doc 2:T0+28->T0+40

Déroulement

T1 Développement d'une base de connaissances

a. Etat de l'art + création d'une base d'attaques et de contre-mesures : phase d'acquisition et d'intégration de données de sécurité dans une base publiquement disponible ; ;
b. Ajout de la notion de patron de sécurité: lintégration des patrons de sécuriténtégration des patrons de sécurité ;
c. Collecte de vulnérabilités, d'attaques liées au matériel. (solution de conception aux problèmes de sécurités récurrents) ;

T2 Analyse de risques :

a. Aide à la mise en place d'une phase d'exigences de sécurité ; : à partir de la base de connaissances, nous inférerons automatique des modèles de menaces ;
b. Mise en place de défenses via des patrons de sécurité , utilisable pour l'Audit de code ;

T3 Aide à l'Audit de sécurité :

a. Aide à la génération de test de sécurité Mise en place d'une phase de test d'intrusion;: aide à la génération de test de sécurité ;
b. Proposition d'un ensemble de techniques pour la sécurisation matérielle. ???????

Quelques Pistes générales?

- Classification vulnérabilités, attaques
- Architecture design, secuiryt pattern pour IOT (-> etudiants)
- Gen. De modèles de menaces (idela pour audit)
- Aides à la génération de tests
- **Inférence de modèles à partir de log ou test + learning** (->elliot)
- Def. De signatures pour reconnaitre IOT malware
- Gen de moniteurs dynamiques pour capturer traces

Quelques Pistes générales?

- Tests sur les données -> def. D'invariants temporels et test d'envoi de mauvaises données
  - Tester la conséquence d'envoyer des mauvaises données sur chaine d'analyse
- Protection gateway par monitoring et learning
- Analyse de composition de devices et proposition de re-composition sur multi-critères
- Analyse code source ou binaire
  - + association avec model learning
- Generateur de regles d'IDS pour gateway

Indutriels ?

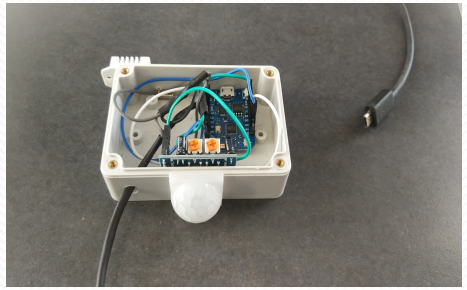
- Qu'attendez vous du projet ?

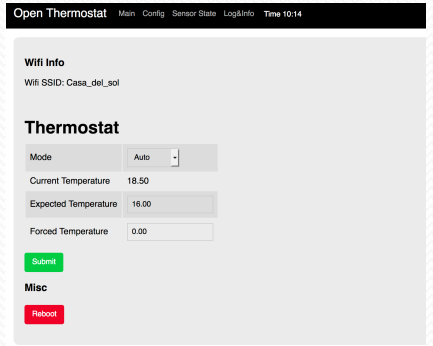
- Transfert de savoir ?
  - Bases ? (CVE ? Etc.) méthodologie ?
- Etude / travail commun ? (stages ?)
- Outils ?

Etude de cas (préliminaire)

Etude de cas

- Objet pour la domotique type smart thermostat
  - https://github.com/sasa27/OpenThermostat



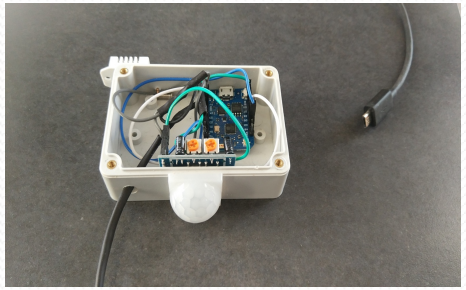


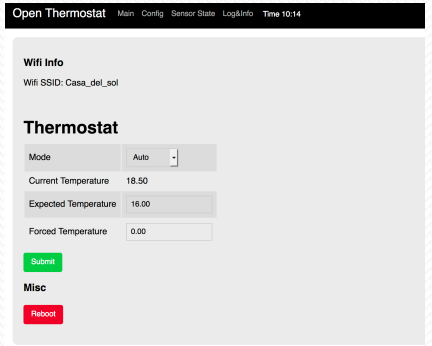
- Protocoles : Wifi, serial, NTP, HTTP
- Interfaces: Web, Rest, serial

- Caractéristiques:
  - gestion de pompes à chaleur
  - configuration Web, gestion heure (requêtes serveur NTP)
  - Plusieurs mode (eco, normal, forcé etc.)
  - Détection de mouvement->plages horaires, capteur température/humidité
  - Envoi de données à un serveur (mode, état, capteurs)

Etude de cas

- Objet pour la domotique type smart thermostat
  - https://github.com/sasa27/OpenThermostat





- Sécurité ? classique
  - WPA2
  - Interface Web Login/mdp
  - Flash firmware Login/mdp

Etude de cas

- Penetration testing + étude de code
  - (manque une analyse du matériel)

- Penetration testing
  - Étude de 15 outils
  - (*ZaProxy*, *CAParser*, Attify, BlackHat arsenal, *kaly linux* (aircrack), hardsploit, *Nmap*, *WebR*, *burpsuite*, *KrackAttack*, sslsuite, BeEF, johntheripper)

- =>besoin d'un minimum d'expertise
- =>tous ne sont pas exploitables directement (moniteur, mapper,etc.)
  - (test matériel: besoin de cartes)

Etude de cas

Penetration testing (+ configuration et écritures de qques tests)

- =>détection de 8 vulnérabilités dont certaines surprises
  - Injections (XSS, SQL, Javascript,etc.)
  - Bruteforce mdp

  - Récupération clée WPA2
  - Problème session ? (1 autre mdp fonctionnel)
  - Appel /reboot avec variables -> reboot avec configuration vierge ???

=>temps nécessaire: 1 heure (ZaProxy) à 1 journée (Burpsuite) par outil (3 personnes)

Etude de cas

- Analyse de code

- retro-engineering -> diagramme de classes et fonctionnalités de chaque classe (5 personnes 1journée/pers)
- 10 classes

- D'autres failles
  - Inputs non validée (outils montrent les conséquences)
  - Code non structuré de façon optimale pour la sécurité
  - Fichier de configuration stocké en clair,
  - mdp par défaut à la première configuration
  - Visibilité des variables en mémoire

  - Partie Rest sans authentification (non détectée par outil, ni par etudiants)

- ->Temps d'analyse très long, pas exhaustif
- Penetration testing insuffisant, analyse suffisante ?

Etude de cas

- Analyse de données
  - Données fournies à Serveur domotique (courbes, données utilisées par autres prog.)
  - aucune vérification (vérification de certains types)
  - Aucune solution, à part validation par des seuils
  - Impact ?

Etude de cas

Solutions ? (en cours)

Modification du code (par chance il est disponible et modifiable)

- Application de security patterns pour éliminer certaines vulnérabilités
- Etude de l'impact des patterns (mémoire, taille sketch, CPU, énergie KO)

- Certains security patterns sont très simples et rapides à mettre en place
  - Ex: session manager
  - Application Firewall -> ok mais très lourd
- D'autres demandent un temps d'analyse long
  - Ex: Single access point (ou le placer ? Champ ?)

=> Temps nécessaire long (1 semaine pour 5 pers. ?)

Etude de cas

Solutions ? (en cours)

- Utilisation de design patterns (etude des patrons du GoF)
  - Memento
  - Composite
  - Etat
  - singleton

- Utilisation de security patterns (étude de 20 patterns)
  - Input guard
  - Single access point
  - Session manager
  - Session timeout
  - Encrypted storage

Etude de cas

Conclusion ?

Temps d'audit (résultats + rapports) et de résolution de pb long (2 sem. / 5 pers.)

Sans code, besoin de nombreux tests à faire à la main

Lors de la conception:

- Utiliser certains patterns pour éliminer certaines vulnérabilités
- Concevoir, implémenter un syst. *auditable* et *vérifiable* tout en garantissant la protection des données (logs visibles pendant audit? Chiffrement à la demande ? Etc.)