

CHPS - AISE - 2022/2023

8 Minutes pour convaincre

Description

Apprendre à découvrir, approfondir et restituer un sujet technique est une compétence aujourd'hui indispensable pour un travail efficace. Il ne suffit pas de réciter tout par coeur, il faut aussi être capable de comprendre. L'objectif de ce projet est de vous approprier un sujet, développer un certain niveau d'expertise pour ensuite le restituer de manière directe et technique. Après avoir choisi un des sujets ci-dessous (à valider avec nous pour éviter les doublons), vous l'approfondirez pour en comprendre tous ses **aspects techniques** (ou presque). Vous en ferez ensuite une restitution de 8 minutes **maximum** pour démontrer votre expertise acquise. Elle sera suivie d'une série de questions. Ce qui est attendu:

- Une présentation "live" en fin de semestre, de **8 minutes**, pas plus !
- Sur une ou plusieurs composantes techniques pointues du sujet choisi
- En vous appuyant *au minimum* sur un **code source** (de vous ou non)
- Possibilité de faire une démo (à inclure dans le temps imparti)
 - *pré-enregistrement recommandé (pour éviter "l'effet démo")*
- **Pas de restriction sur le contenu rendu ou la forme.**

Ne cherchez pas à copier/coller ou résumer des contenus généraux (type page wikipédia ou autre), ce n'est pas ce que nous attendons. Sur certains sujets trop vastes, pas besoin de tout aborder, concentrez-vous sur les points techniques. Pensez au fait que vous pourriez mettre cette maîtrise à votre CV par la suite. Si vous êtes perdus et/ou ne savez pas comment aborder un sujet en particulier, demandez-nous, nous pourrions *peut-être* vous aider :)

Informations importantes

- Par groupe de **2 ou 3**
- Limite de rendu des ressources annexes (codes,...): **Dimanche 8 Janvier 2023, 23h59**
- Date de présentation: **Vendredi 13 Janvier 2023**
- Ordre de passage à définir
- Présentation: **8 minutes max** + 1-2 minutes de questions

Liste des sujets (non exhaustif)

- ☐ Mécanisme du *Thread-Local Storage* (TLS) dans GCC (x86);
- ☐ Ordonnancement temps-réel du noyau Linux;
- ☐ L'architecture ARM;
- ☐ Les compteurs de performance (performance counters) sous Linux en x86;
- ☐ Interception d'appels systèmes via Ptrace (strace, ltrace...);
- ☐ Structure d'un allocateur mémoire (en choisir un);
- ☐ Etude bas niveau du protocole UDP;
- ☐ Fonctionnement du bootloader (BIOS/UEFI...);
- ☐ Le BUS PCI sous Linux, anatomie et fonctionnement;
- ☐ Vue approfondie d'un hyperviseur (virtualisation);
- ☐ Exploitation de vulnérabilités sur la pile (=stack);
- ☐ Débogage et layout mémoire d'un processus (DWARF, ptrace, gdb...);
- ☐ Journalisation du système de fichiers;
- ☐ Amorçage(s) (bootloading) d'un disque;
- ☐ Gestion des processus et threads bas niveau;
- ☐ Extension de fonctionnalités sur compilateur **avec cas d'usage** (GCC plugins);
- ☐ Les mécanismes de sécurité sous Linux (SELinux,...);
- ☐ La mesure du temps sous Linux, résolution et horloges;
- ☐ Étude des conteneurs et des namespaces Linux;
- ☐ *Autre Sujet potentiellement fun (après validation)*