Syllabus
Assembleur 80386

Halim Djerroud
révision: 1.0

Description

Ce cours a pour objectif d'apprendre à programmer et à lire des programmes en assembleur GNU sous un système GNU/Linux en utilisant le jeu d'instructions Intel 80386.

Le cours est divisé en trois parties : (1) Rappel sur les notions de système de numération et d'arithmétique binaire. (2) Organisation interne de la mémoire et du CPU ainsi que les mécanises de communications qui les régissent. Et finalement (3) une présentation approfondie de l'assembleur GNU et les mécanismes de programmation sous-jacente (utilisation de la pile, appels de fonctions, appels système, ...). En bonus, une introduction aux outils d'ingénierie inverse sera présentée à travers un exemple pratique.

L'objectif principal de ce cours est de donner les bases de programmation en assembleur afin de permettre de préparer les étudiants au cours d'ingénierie inverse, analyse des logiciels malveillants, etc.

Je propose que les 24h de cours **Assembleur** soient réparties :
- Cours : 4 séances de 2 heures = 8 heures cours
- TD : 3 séances de 2 heures = 6 séances
- TP : 5 séances de 2 heures = 10 heures

Partie 1 : Système de numération et arithmétique binaire (4 heures)

Objectif :
- Savoir effectuer les conversion dans les différentes bases
- Savoir comment l'information est représentée en mémoire centrale
- Savoir comment l'arithmétique est réalisée
- Savoir utiliser des opérateurs binaires en C
- Savoir identifier les types de variables et leurs tailles

Cours : (2 heure)

- Les systèmes de numération : décimal, binaire, octal et hexadécimal.
- Conversion entre bases
- Arithmétique binaire (signée et non-signée)
- représentation des flottants et la norme IEEE 754
- Représentation des caractères

Contenu du TD : (2 heures)

- Exercices à réaliser sur feuilles sur les conversions binaires
- Manipuler les opérateurs binaires en langage C

Partie 2 : La mémoire centrale (4 heures)

Objectif :

- Connaître l'organisation de la mémoire
- Connaître l'agencement des octets et leur alignement

Cours : (2 heures)

- Organisation de la mémoire centrale
- Caractéristiques d'une mémoire
- Contraintes d'alignement
- Agencement des octets : Endianness (little et big)
- Principe d'implémentation et câblage (facultatif)

Contenu du TD Machine : (2 heures)

- Exercices à réaliser sur feuilles sur les alignements de mots mémoire et le comportement de la pile
- Expliquer le comportement des programmes C et les choix du compilateur GCC

Partie 3 : Le CPU Intel 80386 (6 heures)

Objectif :

- Connaître les différents registres du microprocesseur 80386
- Connaître les instructions du microprocesseur 80386
- Savoir écrire du code en assembleur
- Savoir lire la documentation
- Connaître les modes d'adressage

Contenu du cours : (2 heures)

- Fonctionnement d'un microprocesseur
- Les registres généraux
- Le registre d'état
- Le jeu d'instructions
- Traduire en assembleur les structures conditionnelles et les structures de contrôle
- Les modes d'adressage (adressage absolu, indirect, basé et indexé)

Contenu du TD : (4 heures)

- Exercices sur les registres du processeur
- Écrire des programmes rudimentaires en assembleur sur feuille
- Exercices sur les modes d'adressage

Partie 4 : Assembleur (10 heures)

Objectif :

- Savoir coder en assembleur GNU
- Savoir mettre en place une chaîne de compilation
- Connaître les conventions d'appels de passage de paramètres
- Savoir lire la documentation
- Savoir faire des appels de fonctions de bibliothèque et des systèmes depuis l'assembleur

Contenu du cours : (2 heures)

- Mise en place de la chaîne de compilation (en 32 bits)
- Organisation des fichiers assembleur (sections : text, data et bss)
- Écrire du code assembleur simple (structures conditionnelles et les structures de contrôle)
- Écrire des fonctions en assembleur
- Les conventions GCC appel de fonction assembleur depuis le langage C
- Appeler des fonctions de la *glibc*
- Utiliser les appels systèmes (en 32 bits)

Contenu du TP : (8 heures)

- Mise en place de la chaîne de compilation (32 bits)
- Écrire des programmes à l'aide de GNU AS
- Écrire des programmes en C qui appellent des fonctions écrites en assembleur

Partie 5 : Ingénierie inverse (2 heures)

Contenu du Cours/TP : (2 heures)

- Introduction aux outils ingénierie inverse
- Présentation du logiciel Ghidra à travers un exemple

Prérequis

- Savoir écrire des programmes en langage C et utiliser *glibc* (GNU C Library)
- Savoir utiliser de ligne de commande (bash)
- Savoir utiliser GCC en ligne de commande
- Savoir utiliser d'un éditeur de texte pour écrire du code source (ex : emacs, vim ou autre)
- Savoir utiliser la commande `man`, et identifier les sections
- Savoir différencier un appel système d'une fonction de la bibliothèque standard

— Savoir lire une documentation en anglais