

## 1516 L2 SEPS TDTP 1 - Systèmes d'Exploitation et Programmation Système

### Boot Secteur

Vous devez rendre un rapport à la fin du cours sur Espadon. Assurez-vous que votre rapport porte un nom du type: "1516\_L2\_SEPS\_TDTP\_1\_BootSecteur - *votreNom*". Mettez dans le rapport des copies d'écran des résultats d'exécutions ("Applications" → "Accessoires" → "Capture d'écran") que vous ferez régulièrement le long du TP pour montrer la réussite ou l'échec de vos opérations. Rendez le tout dans un fichier Zip de même nom que précédemment. Pour tous les TP, ne validez votre dépôt sur Espadon qu'une fois le TP totalement terminé; **à la fin de chaque séance, déposez un zip de vos travaux avec votre rapport en l'état, mettez-le à jour à chaque séance, et une fois le TP fini, validez.**

Ce TD/TP est formulé à partir des pages web suivantes, dont la consultation est recommandée:

- <http://a.michelizza.free.fr/pmwiki.php?n=TutoOS.Intro>
- <http://www.websofia.com/2011/10/writing-your-own-bootloader-for-a-toy-operating-system-2/>

Mais nul besoin de commencer par-là, le TDTP y fera référence plus loin.

### I] Connexion

1) Lancez le programme "X2Go" et connectez-vous à une machine Linux "v-ubuntu2016-n" ou n est le numéro qui vous est attribué dans le document "15-16 - SEPS - Attribution machines Ubuntu.pdf".

2) Lancez un terminal ("Applications" → "Accessoires" → "Terminal")

3) Changez le mot de passe de votre machine (avec un mot de passe spécial pour ce cours):

- Tapez la commande "passwd" et suivez les instructions

- Envoyez par email votre nouveau mot de passe à votre enseignant (n'utilisez pas votre mot de passe UPF). Souvenez-vous de ce mot de passe, ou vous serez bloqués ou ralentis par la suite. Gardez le confidentiel, ou quelqu'un pourra se connecter à votre machine quand vous ne l'êtes pas !

4) Vous êtes prêt!

## II] Installation

### 1) Installation Bochs:

Exécutez la commande: `sudo apt-get install bochs bochs-x bochs-sdl`

Dans le répertoire `/home/mathinfo2016/Documents`, créez un répertoire `"SE_and_PS"`, et un sous répertoire `"Bochs"` (commande `mkdir`; ce sera le répertoire de travail).

Créez un document texte et enregistrez le dans `/SE_and_PS /Bochs/` sous le nom `config_bochs.txt`.

Mettez dedans les paramètres suivants:

```
megs: 32
romimage: file="/usr/share/bochs/BIOS-bochs-latest"
vgaromimage: file="/usr/share/bochs/VGABIOS-lgpl-latest"
floppya: 1_44=/dev/loop0, status=inserted
display_library: sdl
boot: a
log: bochsout.txt
mouse: enabled=0
clock: sync=realtime
cpu: ips=500000
```

🔊 - Qu'est-ce la commande `"sudo"` ?

🔊 - Qu'est-ce la commande `"apt-get"` ? Pourquoi l'option `install` est-elle nécessaire ?

🔊 - Expliquez chaque ligne du fichier de paramétrage de Bochs ci-dessus.

### 2) Installez NASM:

Exécutez la commande: `sudo apt-get install nasm`

🔊 - Qu'est-ce que `nasm` ?

### III] TD/TP

#### 1) Ouvrez le code le code du boot secteur qui vous est donné sur Espadon (fichier "bootSect.asm")

🔗 - Modifiez les noms du SE (`iOEM` et `acVolLabel`) à votre nom. Attention à respecter le nombre de caractères.

🔗 - Modifiez également le message de boot `loadmsg` à votre nom.

#### 2) Compiler le code du boot secteur qui vous est donné sur Espadon (fichier "bootSect.asm")

Exécutez la commande: `nsam -f bin -o bootSect.bin bootSect.asm`

🔗 - Qu'est ce que l'option "-f" ? et la "-o" ?

🔗 - Quel est le format de sortie du fichier après compilation ?

🔗 - Vérifiez que votre fichier binaire fait bien 512 octets exactement.

#### 3) Créez une disquette virtuelle avec votre secteur de boot:

Exécutez la commande:

```
cat bootsect.bin /dev/zero | dd of=JmOsFloppy.img bs=512 count=2880
```

🔗 - Qu'est ce que la fonction dd ?

🔗 - Qu'est ce que la fonction "cat" ?

🔗 - A quoi sert le symbole "|" ?

🔗 - Vérifiez que l'image disquette a été créée.

#### 4) Modifiez "config\_bochs.txt" pour booter sur votre image disquette.

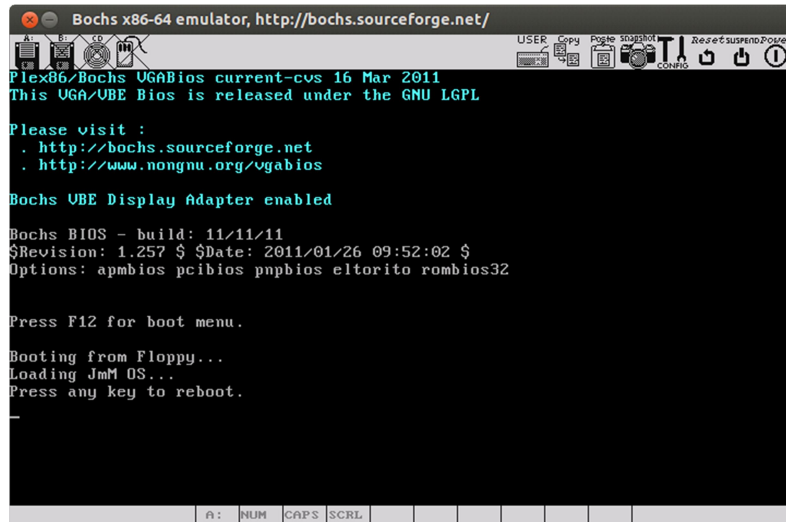
Modifiez la ligne de paramètres adéquate:

```
floppya: 1_44=JmOsFloppy.img, status=inserted
```

### 5) Lancez Bochs :

Exécutez la commande: `bochs -f config_bochs.txt`

☞ - Produisez une sortie d'écran du résultat du boot:



☞ - Pourquoi le code utilisé et tiré de la page "<http://www.websofia.com/2011/10/writing-your-own-bootloader-for-a-toy-operating-system-2/>" est-il plus "intéressant" en premier lieu que le code proposé dans la page "<http://a.michelizza.free.fr/pmwiki.php?n=TutoOS.Intro>" ?

☞ - Pourquoi ces deux codes Assembleur (sur les pages citées) sont ils différents ?

☞ - Essayez de modifier le code pour afficher un second message de boot (de votre choix).

☞ - A quoi servent les nombre "0Dh, 0Ah, 0" placés à la fin des chaines de caractères ?

## IV] Suite

Si vous avez fini le TD/TP avant la fin de la séance, recommencez en suivant les consignes proposées sur le site "<http://a.michelizza.free.fr/pmwiki.php?n=TutoOS.Intro>". Réalisez chaque boot secteur et testez-le, tout en essayant de comprendre chaque étape et de personnaliser le code. Rendez compte de vos essais.