

Projet LAS : Evaluation sur base de tests

Préliminaires

Ces scénarii seront exécutés lors de la défense des projets. **Ils sont susceptibles de changer (légèrement normalement) durant cette dernière semaine du projet.** En cas de changements, vous serez averti via une annonce MooVin.

Préparez 3 terminaux pour pouvoir exécuter facilement les scénarios.

Exécution dans vos programmes

Exécution commandes Linux.

Scénario 1 (simple) : un zombie et un controller

Action	Vérification	OK/KO	Remarques
TEST 1			
Fichier header.h	Voir les ports aléatoires dans header partagé.		
./zombie	Le programme zombie se lance et choisit un port parmi les ports aléatoires. Le zombie affiche le port sur lequel il tourne.		
./controller 127.0.0.1	Le controller se connecte au zombie. Le controller affiche le port sur lequel il a trouvé un zombie.		
ps -u (dans un nouveau terminal)	2 controllers, 1 zombie et un programme inoffensif		
ps -H (dans le controller)	La hiérarchie suivante s'affiche dans le controller : bash zombie bash ps Aucun affichage du côté zombie.		
ls (dans le controller)	Test d'une deuxième commande. Le résultat du ls s'affiche.		
CTRL-D (dans le controller)	Le programme controller et les sous-processus du zombie s'arrêtent mais pas le programme zombie.		
ps -u	Il n'y a plus de sous-processus zombies (bash defunct ou rien) ou controller.		
CTRL-C (dans le zombie)	Le zombie s'arrête.		

TEST 2			
<code>./zombie</code>	Le programme zombie se lance et choisit un port parmi les ports aléatoires. Le zombie affiche le port sur lequel il tourne.		
<code>./controller 127.0.0.1</code>	Le controller se connecte au zombie. Le controller affiche le port sur lequel il a trouvé un zombie.		
<code>curl http://ochoquet.be (dans le controleur)</code>	TOUTE la page html est affichée dans le controller.		
<code>CTRL-C (dans le zombie)</code>	Le zombie et le programme controller s'arrête.		
<code>ps -u</code>	Il n'y a plus de programmes zombies ou controller ou programme inoffensif (bash).		
TEST 3			
<code>./zombie</code>	Le programme zombie se lance et choisit un port parmi les ports aléatoires. Le zombie affiche le port sur lequel il tourne.		
<code>./controller 127.0.0.1 127.0.0.1</code>	Le controller se connecte DEUX FOIS au même zombie. Le controller affiche 2 fois le port (identique) sur lequel il a trouvé un zombie.		
<code>ps (dans le controller)</code>	Le controller affiche 2 fois l'exécution de la commande ps.		
<code>CTRL-D (dans le controller)</code>	Le programme controller et les sous-processus du zombie s'arrêtent mais pas le programme zombie.		
<code>CTRL-C (dans le zombie)</code>	Le zombie et le programme controller s'arrête.		

Scénario 2 (avancé) : Plusieurs zombies et un controller

Action	Vérification	OK/KO	Remarques
TEST 1			
<code>./labo</code>	Deux programmes zombies se lancent et choisissent deux ports aléatoires DIFFERENTS. Les zombies affichent le port sur lequel ils tournent.		
<code>./controller 127.0.0.1</code>	Le controller se connecte aux 2 zombies. Le controller affiche les ports sur lesquels il a trouvé les zombies.		
<code>ps -u</code> (dans un nouveau terminal)	2 controllers, 2 zombies et 2 programmes inoffensifs		
<code>ps -H</code> (dans le controller)	La hiérarchie suivante s'affiche 2 FOIS dans le controller : bash labo zombie bash ps zombie bash ps Aucun affichage du côté zombie.		
<code>CTRL-D</code> (dans le labo)	Tout s'arrête.		
<code>ps -u</code> (dans un terminal)	Plus aucun processus controller, zombie ou encore programme inoffensif !		

Note (20 pts) :

Autres Remarques :