

3. Bugs

Bonjour, bienvenue dans cette troisième séquence de la programmation qui est consacrée aux bugs. Vous en avez sûrement tous rencontré, je vais vous en raconter quelques-uns.

Le premier, c'est l'USS Yorktown, qui est un bateau américain de guerre qui a été lancé en 96 avec un programme Navy's Smart Ship qui signifie "bateau intelligent". Lors d'essais, un membre d'équipage s'est trompé sur ce qu'il rentrait dans la machine, comme valeur d'entrée, il a rentré un 0 sur une valeur qui en théorie n'était pas censée être 0. La conséquence, c'est qu'il y a eu dans le programme une division par 0 qui est remontée et qui a complètement planté le système d'exploitation, et la conséquence de ça c'est que plus rien ne marchait sur le bateau, en particulier pas les moteurs, ce qui est quand même un peu surprenant, et donc le bateau était complètement planté au milieu de l'eau. Vous pouvez voir un peu plus de détails sur cette page web, et donc ce qu'il faut retenir, c'est qu'il faut tester les valeurs d'entrée parce qu'on n'est jamais sûr que l'utilisateur final est bien conscient qu'il y a des valeurs interdites. Ce genre de valeur non attendue est également une porte d'entrée pour les attaques sur les programmes.

Un autre exemple, c'est le lanceur de satellites Ariane 5. Ce lanceur de satellites est le successeur d'Ariane 4, qui marchait très bien avec un programme. Et en fait ils ont repris le programme d'Ariane 4 pour le mettre dans Ariane 5. Simplement, Ariane 5 est beaucoup plus grosse, beaucoup plus lourde qu'Ariane 4, et donc il y a des valeurs maximales qui étaient correctes pour Ariane 4 qui n'étaient plus correctes pour Ariane 5. Donc au premier lancer d'Ariane 5, ces valeurs maximales ont été dépassées, et en informatique quand on dépasse les valeurs maximales autorisées d'un entier ou d'un flottant, il se passe des choses bizarres, en particulier pour les flottants, les nombres à virgule flottants, on peut avoir des valeurs très grandes qui deviennent très petites. On peut sur les entiers, avoir des entiers positifs qui deviennent négatifs, et donc tout ça a fait qu'il y avait une valeur complètement absurde et le lanceur de satellites a commencé à partir sur le côté à cause de cette mauvaise valeur, et il a fallu le détruire pour ne pas qu'il tombe sur les habitations. Donc la conclusion de ça, c'est qu'il faut faire attention aux variables qu'on utilise et à leur type pour être sûr qu'on ne va pas dépasser les valeurs maximales et minimales autorisées.

Alors qu'est-ce que c'est qu'un bug ? En fait un bug c'est quand l'utilisateur au final n'a pas ce qu'il veut. Il y a tout un tas de raisons pour lesquelles on peut avoir un bug. Ça peut être une erreur de programmation, c'est-à-dire qu'effectivement la personne qui a programmé s'est trompée. Oublier un cas particulier et faire une typo, avoir un dépassement de capacités comme ce dont je viens de parler, un accès illicite à la mémoire, c'est-à-dire qu'on accède à un morceau de mémoire qui n'était pas prévu et donc qui n'était pas initialisé avec une valeur correcte, ou tout un tas d'autres erreurs.

On a aussi un autre type de bug qui est plus sournois, mais finalement au moins aussi fréquent, qui est l'erreur de communication, c'est-à-dire qu'effectivement, entre la personne qui a demandé le programme et la personne qui l'a programmé à la fin, il y a eu une mauvaise compréhension sur les valeurs d'entrée, sur ce que devait faire le programme, sur tout un tas de choses.

Il peut aussi y avoir une erreur de communication entre 2 programmeurs, parce que les programmes actuels ne sont pas écrits par une seule personne, ils sont écrits par des milliers de personnes qui doivent interagir, ils doivent comprendre ce que fait le morceau de programme des autres personnes.

Pour vous montrer qu'effectivement les programmes sont très gros, je vous ai montré un petit exemple avec un dessin qui représente le nombre de lignes de code d'un certain nombre de programmes. Vous pouvez retrouver ce dessin en meilleure résolution à cette adresse web, et donc je vais vous en expliciter quelques-uns. Un pacemaker c'est à peu près 100 000 lignes de code, donc vous voyez c'est déjà difficile à écrire par une seule personne et ce n'est qu'un petit programme par rapport à ce dont je vais vous parler. Après, le télescope spatial Hubble, à l'intérieur il y a 2 millions de lignes de code, donc c'est déjà beaucoup. Un autre exemple de taille supérieure, c'est à peu près 5 millions de lignes de code, ce qui correspond aux serveurs de World of Warcraft, ou du code embarqué dans un Boeing 787. Alors encore plus gros, vous avez Facebook qui représente à peu près 60 millions de lignes de code. Donc vous imaginez bien que c'est très compliqué sur 60 millions de lignes de code, d'être sûr qu'il n'y a pas une erreur qui s'est insérée quelque part, et c'est pour ça que vos programmes de temps en temps, ils ne donnent pas leurs bons résultats, ou ils plantent.