

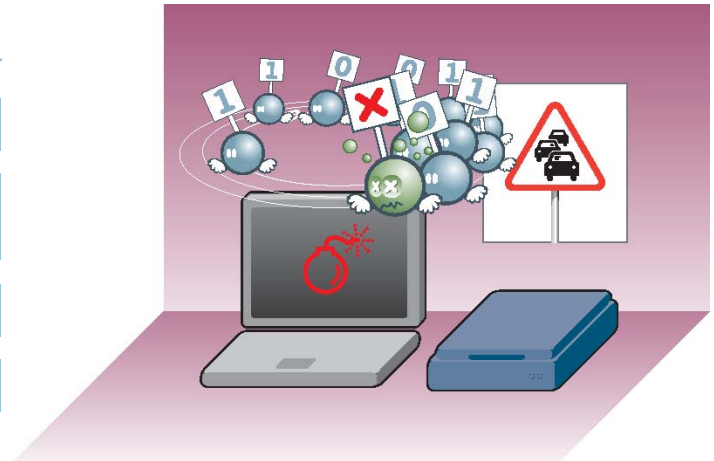
Résumé

Actuellement, des milliers de logiciels tournent sur des millions des machines dans le monde. Des bogues peuvent exister au sein du code des logiciels ou systèmes opératoires. Si les producteurs de logiciels ne décident pas de prendre le problème au sérieux, on

peut considérer que l'exploitation de ces vulnérabilités va continuer à se développer et que les dégâts vont augmenter de jours en jours. Le but de cette fiche est d'expliquer ce que sont les bogues et comment ils mettent en danger les systèmes informatiques.

Table des matières

- 1 C'est quoi ? →
- 2 Est-ce que la programmation suit des standards de sécurité ? →
- 3 Pourquoi se protéger ? →
- 4 Comment se protéger ? →



1 C'est quoi ?

Un bogue (en anglais -«bug») est une faiblesse fortuite au sein d'un logiciel ou d'un «hardware» (matériel) pouvant mettre en danger le bon fonctionnement du système informatique. Il y a lieu d'inclure sous le terme «hardware» les microcodes et autres codes embarqués dans les circuits et puces électroniques.

On appelle déboguer (en anglais - «debugging») le fait de rechercher, de corriger et d'éliminer les bogues.

1.1 Quelques informations générales

1.1.1. L'ORIGINE DE CE NOM

Le premier ordinateur utilisé aux Etats-Unis à la fin de la seconde guerre mondiale n'avait pas vraiment le même aspect que nos machines actuelles. A cette époque, les machines étaient beaucoup plus volumineuses et dégageaient beaucoup de chaleur. Elles étaient donc des abris privilégiés pour beaucoup d'insectes (en anglais, insecte se dit «bug»). La légende raconte qu'une des premières pannes de ce type de machine était due à un insecte se promenant sur les circuits au cœur du premier ordinateur.

En français, on utilise le mot bogue (au masculin) en faisant référence à l'enveloppe de la châtaigne qui porte des piquants et l'empêche de tourner rond. On parle de bogue pour qualifier quelque chose qui ne tourne pas rond, quelque chose qui ne fonctionne pas comme prévu et il faut déboguer pour en trouver la raison.

1.1.2. STATISTIQUES

- ➔ Selon le CERT/CC (Computer Emergency and Response Team/Coordination Center) aux Etats Unis, en 1988 seuls 8 bogues avaient été recensés.
- ➔ En 1998, 10 ans après, on en dénombrait 3.784.
- ➔ En 2002, selon le même organisme, on en recensait plus de 82.000.

Cette croissance est bien sûr à mettre en parallèle avec l'augmentation du nombre des machines et logiciels, mais aussi avec l'ouverture du monde informatique vers un accès ouvert via Internet.

1.1.3. LE BOGUE DE L'AN 2000

Le bogue ayant connu la plus grande couverture médiatique est certainement celui de l'an 2000. En effet la plupart des codes développés ou embarqués durant les années 80 stockaient les années sur base les deux derniers chiffres (l'année codée sous la forme 80 signifiant pour l'ordinateur l'année 1980). Le risque était que lors du passage à l'an 2000, le système informatique ne se croit revenu en 1900 et simule un retour en arrière de 99 années, considérant alors l'année 2000 considérée dans la mémoire de l'ordinateur sous la forme 00 ce qu'il pourra interpréter comme étant l'année 1900.

[→ suite](#)

La prévoyance des fabricants, alliée à la bonne gestion de la plupart des systèmes informatiques a limité les impacts de ce phénomène, mais il est certain que sans les précautions prises, la fiabilité des systèmes informatiques aurait été mise à mal.

1.2 Des bogues très fréquents

De nombreux bogues sont la cause de la plupart des attaques de pirates informatiques. En effet, leur exploitation peut causer des dommages importants au sein d'un système informatique. Il est malheureux de constater que ces faiblesses ne sont rien d'autres que des erreurs de conception ou de programmation.

Les vulnérabilités les plus connues et les plus exploitées sont les suivantes :

1.2.1. «BUFFER OVERFLOW» OU «HEAP OVERFLOW»

Ce bogue est certainement le plus exploité. Celui-ci permet, dans certains cas, des comportements non prévus pas les développeurs du programme qui peuvent permettre à un pirate d'obtenir des droits et privilèges particuliers sur le système vulnérable. Il s'agit en fait d'un dépassement de mémoire tampon allouée, qui va être traité et exécuté par le logiciel.

PAR EXEMPLE : Un utilisateur est sensé introduire 10 caractères dans une zone de saisie sur une page Web. Une personne mal intentionnée, introduit les 10 caractères mais fait suivre ceux-ci d'une chaîne d'autres caractères. Si le logiciel n'est pas bien conçu, cette chaîne de caractères risque d'être, sous certaines conditions, exécutée.

Selon le CERT, le 25 Janvier 2003 «Slammer Worm» a infecté 90 % des cibles potentielles en 10 minutes. Cette attaque exploitait une faiblesse de type «Heap overflow» liée aux services de Microsoft Windows SQL2000 Server Resolution Services (SQL = Structured Query Language). Le bogue étant «caché» derrière beaucoup d'autres logiciels, de nombreux responsables informatiques n'étaient pas au courant de leur exposition à ce risque.

1.2.2. «FORMAT STRING»

Cette vulnérabilité était présente dans la plupart des machines et logiciels depuis de nombreuses années. Toutefois, ce n'est qu'au milieu de l'année 2000 qu'on attira l'attention sur elle. Elle est le résultat de l'utilisation de fonctions inappropriées pour le traitement de chaînes de caractères. L'apparition des logiciels Microsoft Windows NT4 et 2000, a révélé cette faiblesse. En effet, ces deux logiciels ne traitaient pas les données correctement et permettaient même de créer des portes d'accès dérobées (backdoor).

1.3 Les bogues «web» ou «web bugs»

Le terme bogue est également couramment utilisé pour qualifier différents mécanismes liés aux sites Web. Toutefois, il faut bien préciser que le terme bogue est dans ce cas détourné de sa définition initiale. En effet, les mécanismes repris sous cette catégorie sont sciemment mis en place et il ne s'agit en rien d'une erreur fortuite, mais bien d'une fonctionnalité développée dans un but précis. Ces termes sont développés ci-après.

1.3.1. DÉFINITION DE BOGUES «WEB»

Il s'agit en fait de mécanismes cachés dans des sites Web dans le but de récolter des informations sur l'utilisateur lui-même (adresse IP, version du browser, ...) ainsi que sur les sites Web qu'il a précédemment visité. Ces informations sont ensuite envoyées à un tiers et ce dans un but d'espionnage pur et simple ou dans un but de ciblage commercial.

Une technique couramment utilisée

Selon une analyse publiée par «Intelytics» parmi 51 millions de sites Web inspectées, 16 millions contenaient un mécanisme d'espionnage, soit 31,4 % des sites.

Ces mécanismes sont habituellement cachés derrière des images d'apparence inoffensives, des petits exécutables, des scripts, des «cookies» et des applications. Ces bogues peuvent être situés sur des sites Web, des messages, des «newsgroups»,

2

Est-ce que la programmation suit des standards de sécurité ?

2.1 Langage de programmation et bogue

La plupart des langages de programmation utilisés actuellement sont devenus très performants et permettent la mise en place d'applications complexes offrant des services de type transactionnel sur un réseau public.

Ces langages de programmation prennent en charge la vérification de la syntaxe du langage mais ne prennent pas tous en compte le contrôle d'erreurs du type traitement des «buffer overflow».

L'utilisation basique de ces langages ne tient nullement compte de la sécurité, et il faut absolument prévoir une face de «blindage» de ces applications. Que l'on utilise des langages tels que C, C++, Java, Perl ou autres Visual Basic, il est indispensable de passer par une phase de mise à l'épreuve du codage, de manière à déjouer toutes les faiblesses, ce qu'on appelle la phase de test de correction des bogues.

[suite au verso →](#)

→ suite

2.2 Les solutions « open source » et bogue

Il existe dans le monde informatique un débat sans fin sur la sécurité des logiciels « ouverts » qui donnent accès à leurs sources de codage en comparaison avec le monde « fermé » qui ne donne pas cet accès.

Les défenseurs du monde fermé se basent sur le fait qu'il est plus difficile d'exploiter les faiblesses d'un ennemi inconnu. Les défenseurs du monde ouvert argumentent que le programmeur met en jeu sa réputation à chaque ligne de programmation.

La réalité statistique est que les deux mondes sont vulnérables et que la justification d'une solution « Open Source » ne peut pas se faire du point de vue de la sécurité.

2.3 Les responsables et bogue

Il n'y a actuellement aucun moyen de tenir le fabricant d'une solution logicielle pour responsable de l'exploitation de bogues dans son produit et ce n'est qu'au début des années 2000 que l'on a vu apparaître des sanctions contre les personnes mal intentionnées qui exploitent ces vulnérabilités.

Il semble qu'une solution serait de soumettre les producteurs de solutions au respect de standards de sécurité avant d'autoriser la commercialisation de leurs solutions. Le respect de ces standards pourrait être vérifié par des laboratoires indépendants.

Il peut toutefois être intéressant d'analyser la position de quelques grands acteurs du marché vis à vis de cet aspect de sécurité.

2.3.1. MICROSOFT

On a tendance à penser que les produits Microsoft sont plus bogués que les autres. Il faut toutefois prendre la peine de pondérer les chiffres sur base du nombre d'implémentations. Il est évident que sur ce point Microsoft est victime de son succès.

Au début des années 2000, Bill Gates a déclaré que la sécurité des logiciels serait le cheval de bataille pour les 5 prochaines années. Il est actuellement un peu tôt pour juger des résultats. Toutefois, on peut remarquer que certains logiciels n'ont pas été publiés à la date prévue car soumis à des validations au niveau de la sécurité et que la plupart des lignes de codes de Microsoft Windows ont été réécrites pour satisfaire aux nouvelles exigences.

2.3.2. ORACLE

Au cours du printemps 2003, le CERT/CC (aux Etats Unis) a révélé 37 vulnérabilités au sein du système de base de données d'Oracle. La majorité de ces vulnérabilités étaient situées au sein du codage même du produit. Cela a mis à mal l'argumentation commerciale de fiabilité avancée jusqu'alors par le fabricant.

2.3.3. LINUX

Comme mentionné plus haut, ce n'est pas parce que l'on travaille dans le monde « ouvert » que ces logiciels sont exempts de tout bogue. Selon la communauté « Open Source » elle-même, le gain en popularité de Linux va entraîner l'augmentation des bogues et ce parce que ses logiciels vont intéresser de plus en plus le monde des pirates informatiques qui vont traquer des bogues encore inconnus.

De plus, il est probable que cela va entraîner la mise en place de logiciels non protégés, car l'intérêt accru pour ces logiciels n'est pas automatiquement lié à une maîtrise technique de la part des utilisateurs.

4 Pourquoi se protéger ?

Parce que les bogues constituent une menace considérable pour tous les utilisateurs des systèmes informatiques, il convient de se protéger.

En effet, ils peuvent causer des pertes importantes :

▲ pertes financières directes

- destruction de données cruciales,
- mise hors service de tout le système informatique,
- ...

▲ perte de réputation

- mise en cause de la crédibilité dans le cas de divulgation d'informations confidentielles,
- ...

▲ perte de temps

- efforts produits pour restaurer les données détruites,
- ...

[→ suite](#)

4 Comment se protéger ?

La lutte contre l'apparition de ces vulnérabilités ne se situe malheureusement pas au niveau des utilisateurs mais bien au niveau des producteurs des logiciels ou systèmes informatiques. Dans ce cadre, pour les utilisateurs, on ne peut parler que de mesures de protection réactives et non pro-actives.

4.1 La mise à jour des logiciels et systèmes informatiques

Il existe sur Internet différents sites Web tels que ceux du SANS (SysAdmin, Audit, Network, Security institute), CERT et de nombreux autres qui publient périodiquement la liste des principales vulnérabilités ainsi que les mesures à prendre pour colmater les « brèches ».

Les administrateurs des systèmes informatiques doivent se tenir au courant des vulnérabilités et d'appliquer les correctifs dans les plus brefs délais. Il est toutefois conseillé de valider les correctifs sur une plate-forme de test afin d'éviter tous problèmes de compatibilité avec les logiciels en production.

4.2 La validation des systèmes informatiques

Différentes entreprises de conseil informatique peuvent délivrer des services dits « Attack and Penetration Tests » dont le but est de valider la globalité de l'infrastructure en se mettant tantôt dans la peau d'un attaquant externe, tantôt dans la peau d'un utilisateur standard du système informatique. Les mêmes entreprises font suivre ce type de mission par un service de veille technologique, permettant ainsi de conserver les acquis de la mission initiale.

En ce qui concerne les bogues « web », le problème se situe à un autre niveau. Il est important de faire la balance entre la prise de risques et la limitation des fonctionnalités. En effet, la plupart des mesures de sécurité possibles entraînent des pertes de fonctionnalités sur Internet.

Toutefois, il vous est conseillé de suivre les conseils suivants :

4.2.1. LIMITATION DES TÂCHES DE FOND

Il est possible de bloquer l'exécution de scripts ainsi que l'usage de cookies au niveau de la configuration du « navigateur » (browser) utilisé.

4.2.2. UTILISATION DE LOGICIELS DE DÉTECTION

Il existe sur Internet des outils de détection et de nettoyage de bogues « web ». Il est assez instructif de faire tourner ce type de logiciel sur la machine et de voir le nombre de modules d'espionnages actifs. Le plus connu actuellement est le logiciel de détection et de nettoyage « Ad-Aware », publié par l'entreprise Lavasoft.