1. Introduction générale :

Un code malveillant est un programme qui a pour but de s'introduire dans un système informatique, de l'endommager ou d'en tirer profit de quelconque manière. Ce terme général couvre toutes sortes de logiciels indésirables comme les virus, les vers, les chevaux de Troie, les portes dérobées et autres espiogiciels ou publiciels .

Ce chapitre explique comment fonctionnent les différents types de codes malveillants et comment s'en protéger.

2. Classification :

2.1. Virus :

Un virus est un *code malveillant* qui **se propage à l'aide d'un autre programme** ou fichier qui joue le rôle d'**hôte** du virus et sans lequel le virus ne peut pas se propager.

Par définition un virus a besoin d'un hôte pour se propager.

Exemple de hôte :

Fichiers exécutables. La forme la plus classique d'un virus infecte des fichiers exécutables comme les fichiers .**exe** ou .**com** sous Microsoft Windows➔ Ils s'exécutent lorsque le programme est lancé.

2.1.1. Signature d'un virus :

Les virus sont programmés pour ne pas infecter plusieurs fois un même fichier. Ils intègrent ainsi dans l'application infectée une suite d'octets leur permettant de vérifier si le programme a préalablement été infecté : il s'agit de la **signature virale**.

2.2. Vers :

Un ver informatique est un programme qui peut **s'auto-reproduire** et **se déplacer à travers un réseau** en utilisant les mécanismes réseau, *sans avoir réellement besoin d'un programme hôte* (fichier, secteur d'amorçage d'un support de stockage ,etc.) pour se propager ➔ **un ver est donc un virus réseau**.

2.2.1. Comment se protéger ?

Il est simple de se protéger d'une infection par ver. La meilleure méthode consiste à ne pas ouvrir "à l'aveugle" les fichiers qui vous sont envoyés en fichier attachés.

2.3. Cheval de Troie :

Un cheval de Troie est un programme indépendant qui parait avoir une fonction utile et légitime (logiciel de chat, application mathématique, ...) alors qu'il contient des fonctionnalités malveillantes.

Quand un utilisateur exécute le cheval de Troie, les fonctions malveillantes sont aussi exécutées.

2.3.1. Pourquoi un cheval de Troie ?

- Installer une porte dérobée dans la machine de la victime.
- Installer un spyware.
- Modifier l'implémentation d'un champ (le champ d'authentification d'un logiciel)➔ insérer un instruction (password.getString() + envoyer le résultat par email au pirate)
- Etc...

2.4. Backdoors :

Une *backdoor*, ou porte dérobée, est un logiciel qui permet à un attaquant de prendre contrôle d'un ordinateur à distance.

2.4.1. Comment et pourquoi ?

Les *backdoors* les plus simples ouvrent un port TCP ou UDP (sur la machine de la victime) et attendent que l'attaquant s'y connecte, et envoie des CMD sous forme de texte (le texte sera interpréter (sur la machine de la victime) en utilisant des conditions ➔ EX : if CMD.equale(‘’ lister‘’) ➔ lister l'arborescence, et l'envoie au pirate).

Les fonctionnalités de base d'une *backdoor* permettent de pouvoir **copier des fichiers sur la machine infectée** et d'en **lancer l'exécution, etc...**

2.4.2. Comment se protéger ?

Il suffit d'installer un firewall, c'est-à-dire un programme filtrant les communications entrant et sortant de votre machine.

2.5. Spywares (espiogiciel) et adwares (publiciels)

Un *spyware*, ou espiogiciel, est un logiciel qui recueille des informations sur la machine infectée et les transmet au pirate.

Les *adware*, ou publiciels, présentent des publicités à l'utilisateur sans pour autant l'espionner.

2.6. RootKits

Un *rootkit* est un ensemble d'outils qui permet de cacher la présence de logiciels malveillants sur un ordinateur infecté.

2.6.1. Comment ?

- Le cacher parmi les fichiers système ➔ sera chargé avec le système
- Utiliser une signature d'un driver connu par le système d'exploitation.
- Insérer le code malveillant dans un driver qui existe dans le système d'exploitation.

2.6.2. Comment se protéger ?

Les systèmes d'exploitation se protègent contre les *rootkits* en vérifiant la signature et l'intégrité de tous les éléments critiques d'un système.

2.7. Cryptovirus (polymorphisme)

Le code malveillant est chiffré avec une clef aléatoire et différente pour chaque copie du code malveillant. Une petite routine de déchiffrement utilise la clef pour recréer le code original.

Objectif : rendre plus difficile a détecter la signature : Le polymorphisme permet de modifier un code malveillant à chaque infection, de manière à ce que chaque nouvelle copie soit différente de la précédente.

3. Protections :

3.1. Logiciels antivirus :

Les antivirus ont 2 modes de fonctionnement de base : **scanner la signature** et **l'analyse comportementale.**

Contremesure d'un antivirus

- La suppression du code correspondant au virus dans le fichier infecté ;
- La suppression du fichier infecté ;
- La mise en quarantaine du fichier infecté, consistant à le déplacer dans un emplacement où il ne pourra pas être exécuté.

3.1.1. Signature :

Les antivirus s'appuient ainsi sur cette signature propre à chaque virus pour les détecter. Il s'agit de la méthode de recherche de signature (scanning), la plus ancienne méthode utilisée par les antivirus.

Cette méthode n'est fiable que si l'antivirus possède une base virale à jour, c'est-à-dire comportant les signatures de tous les virus connus.

3.1.2. Analyse comportementale.

Une autre méthode pour détecter si un programme est un code malveillant est de le laisser s'exécuter et de contrôler ce qu'il fait. Évidemment, l'exécution doit être simulée pour éviter sa propagation.

3.2. Architecture réseau sécurisé (chapitre4)

Une protection efficace ne peut être obtenue que si on installe des logiciels antivirus à tous les niveaux du réseau :

- sur les postes de travail et ordinateurs portables.
- sur les serveurs de fichiers.
- sur les serveurs de messagerie.
- finalement sur les proxys HTTP, FTP et SMTP.