Lutter contre les codes malveillants

Valérie Viet Triem Tong

Lutter contre les codes malveillants

V. Viet Triem Tong*
*CentraleSupélec, Inria, Université de Rennes 1, CNRS, IRISA UMR 6074,
F-35065 Rennes, France

Aujourd'hui, trois quart des habitants de la planète possèdent un téléphone portable. Ce téléphone portable est en réalité un tout petit ordinateur doté de plus de capacités de calcul et de stockage d'information qu'un ordinateur de bureau des années 1990. Ces téléphones rendent plus de services que leurs ancêtres, ils peuvent prendre des photos, se connecter à internet, tenir un emploi du temps. Mieux encore, ils peuvent exécuter des applications choisies par l'utilisateur sur des marchés d'applications comme GooglePlay ou l'App Store. Ces applications sont des programmes informatique écrits par des développeurs qui peuvent appartenir à des entreprises informatiques mais aussi de simples particuliers. Bref, tout le monde peut écrire, et mettre à disposition de tous, une application pour téléphone portable.

Le développeur gagne de l'argent lorsque cette application est téléchargée, si celle ci est payante. Si par contre l'application est gratuite, le développeur se rémunère par la publicité intégrée dans l'application ou via des achats intégrés. La note est peu élevée pour l'utilisateur (de l'ordre de quelques euros) mais les gains montent vite pour le développeur si l'application est populaire puisqu'elle peut alors avoir des millions d'utilisateurs.

Attirés par l'argent facile, certains développeurs peu scrupuleux sont tentés d'ajouter du code malveillant à leur application voire même à celles proposées par d'autres. Ces codes ont pour but de faire gagner encore plus d'argent au développeur malveillant. Pour cela, ces codes malveillants peuvent envoyer automatiquement des messages à des numéros surtaxés. Ils peuvent neutraliser le téléphone de l'utilisateur pour l'obliger obéir aux commandes de l'attaquant. L'ensemble des téléphones (ou tablettes) ainsi neutralisé par l'attaquant forme un réseau de machine qu'il peut utiliser pour réaliser des attaques informatiques. Enfin ces codes peuvent exiger de l'argent pour rendre à l'utilisateur le contrôle de ses propres données.

La guerre est déclarée.
La prolifération des codes malveillants occasionnent une guerrilla informatique entre plusieurs acteurs : les auteurs de ces codes qui souhaitent continuer leurs activités, les éditeurs de produits de sécurité et les responsables des marchés d'applications qui veulent être capables de détecter et contrer le plus rapidement possible ces codes malveillants.

Les combattants fourbissent leurs armes. L'attaquant écrit son code. Souvent ce code contient plusieurs parties: prise de contrôle du téléphone, réalisation de la malveillance et protection du code malveillant lui-même.

Les défenseurs examinent tous les codes qui sont mis sur le marché à la recherche d'éventuels codes malveillants. Ce travail ressemble à celui qui consiste à chercher une aiguille dans une botte de foin : quelques dizaines de lignes de codes malveillants cachées dans d'innombrables lignes de code bénin.

Le temps joue avec l'attaquant. Plus les défenseurs mettent de temps à détecter un code malveillant plus celui-ci peut s'executer en toute impunité.

Première ligne de défense: repérer les codes malveillants connus. Les défenseurs gardent une banque de codes déjà identifiés comme malveillants. Dans ces banques de données, les codes malveillants sont identifiés par certains traits particuliers que l'on appelle des signatures. Ces signatures sont les plus petites possibles pour que la recherche soit envisageable sur une grosse quantité de code. La première ligne de défense est alors de vérifier rapidement que le code étudié n'est pas déjà présent dans ces banques, c'est à dire que le code étudié ne présente pas de signature connue. Cette approche est peu couteuse car il existe des moyens algorithmiques rapides pour ce travail. Elle nécessite cependant de maintenir des banques de données à jour et bien organisées.

Contournement de la première ligne de défense: écrire un code inconnu. Cette première ligne de défense, c'est un peu la ligne Maginot : elle est très efficace mais l'attaquant peut la contourner. Pour cela, il suffit à l'attaquant d'écrire un code qui n'affiche aucun trait particulier lié aux codes malveillants déjà connu. Son nouveau code ne présente alors aucune signature et il n'est alors pas reconnu comme malveillant.

Deuxième ligne de défense: évaluer la dangerosité du code. Lorsque le défenseur étudie un code inconnu, il peut vérifier que toutes les exécutions possibles avec ce code offre des garanties de sécurité. Pour cela, nul n'est besoin d'exécuter le code: le défenseur fait appel à un analyseur statique qui lit et étudie toutes les exécutions possibles. Un analyseur statique peut être un être humain qui va lire et comprendre le code, ou bien un autre programme informatique remplissant cette fonction. Si l'analyseur est humain il va pouvoir faire une étude complète et très fine du code malveillant mais cela lui demandera beaucoup de temps et il ne pourra pas en étudier beaucoup. Si l'analyseur est un programme, l'analyse sera plus rapide mais moins performante.

Contournement de la deuxième ligne de défense: écrire un code incompréhensible donc non-évaluable. Bien sur, l'attaquant connaît les armes à la disposition du défenseur et sait donc que son code sera étudié par un analyseur statique. Pour contrer l'analyse statique, l'attaquant va donc écrire un code auquel l'analyseur ne pourra accéder ou qu'il ne pourra pas comprendre et qui le fera échouer. Pour cela, il dispose de moyens relativement simples comme cacher le code malveillant dans des ressources de l'application comme des images ou du son, il peut aussi le chiffrer ou encore le rendre disponible uniquement à l'exécution.

Dernier rempart: exécuter et observer le code. Les méthodes de défense précédentes se focalisent essentiellement sur la forme et la signification du code malveillant lui même. Ces méthodes sont déjouées par l'attaquant car il lui est possible de maquiller un code malveillant, autrement dit de cacher sa forme et le rendre incompréhensible pour l'analyseur. Lorsque ces méthodes ont échoué, il reste au défenseur à essayer d'exécuter le code qu'il considère comme suspect. Son objectif est alors d'observer le comportement du code. Lorsque le défenseur exécute un code malveillant, ou tout du moins suspect, il prend un risque plus important que lorsqu'il se contente de l'analyser statiquement. Le défenseur va exécuter un code qu'il ne maîtrise pas dans sa propre infrastructure. Le défenseur va devoir prendre des précautions de sécurité car cela revient peut être à faire entrer le loup dans la bergerie. En effet, le code est peut être plus dangereux que prévu et pourrait attaquer le défenseur lui-même. Le défenseur exécutera donc ce code sur un plateforme isolée et protégée comme un biologiste utiliserait une salle blanche. Néanmoins, le fait d'exécuter le code est un avantage important pour le défenseur car cela rend caduque tous les moyens de maquillage de l'attaquant cités jusque là. Ce qui devient important dans cette dernière approche est que l'on étudie ce que fait le code et non plus comment ce code est écrit.

Dernier atout de l'attaquant: faire que le code malveillant ne se déclenche pas sur demande chez le défenseur. Le dernier rempart du défenseur est encore contournable par l'attaquant. Ce dernier rempart repose en effet sur le fait que le défenseur va être capable d'exécuter un code malveillant. Pour faire échouer cette dernière approche, les attaquants vont développer des attaques furtives qui sont difficiles à observer. Le but des attaquants est de s'assurer que leur code malveillant ne s'exécute pas dans une salle blanche en laboratoire mais bien sur l'appareil d'un utilisateur. Pour cela, ces codes vérifieront qu'ils ne s'exécutent pas sur un émulateur, ou bien sur un réseau isolé d'Internet. Ils attendront aussi un évènement leur montrant qu'ils sont bien en face un utilisateur comme la reception d'un SMS, la décharge de la batterie. Certains de ces codes malveillants ne s'exécutent qu'à la demande de l'attaquant, rendant ainsi quasi-impossible l'observation de leur comportement par le défenseur.

Que faire maintenant ?

Il est difficile de quantifier la partie des codes malveillants qui sont effectivement détectés par les défenseurs puisqu'il existe sans doute des codes malveillant qui ne seront jamais détectés. L'avantage est à l'attaquant, mais les efforts que déploient les défenseurs sont récompensés puisqu'ils sont capables de détecter une quantité important de codes malveillants. Pour qu'un code malveillant agisse longtemps dans l'ombre il faut déployer beaucoup de savoir faire qui n'est pas à la portée de tous les développeurs.