GOUVERNEMENT

Liberté
Égalité
Fraternité

Pôle d'expertise de la régulation numérique

Mobile applications: what are the security issues for their distribution outside OS stores?

The democratization of smartphones has brought with it radical changes in the way we install our applications. From a decentralized and open model, with downloads from third-party sites on our computers, we have moved to a centralized and sometimes closed model on our smartphones. Now, installation of applications is often, and in some cases exclusively, done from the application store of the operating system developer.

This restricted distribution method is being challenged by several national and international regulations under development, including the Digital Market Act, which aims to open up the distribution of applications to third-party sources. While often justified by economic stakes, such opening raises issues related to user security and protection. This note presents the technical considerations related to these issues and how to address them.

Shedding light on...

February 2022

.02

When it comes to the issue of applications distribution on mobile operating systems, there are a number of perspectives to consider. On one hand, operating system developers tend to prefer to distribute applications via exclusive application stores in order to protect the user from malicious applications that could compromise the security of the hardware or the user. On the other hand, legitimate application publishers may prefer not to be constrained by the pricing or editorial policies of exclusive application stores and would therefore like more competition in this area, including the possibility of distributing their applications directly themselves. Regardless, it is important to be able to offer the user both the best protection against malware, and also the most choice and competitive pricing for the applications they might download.

WHAT IS MALWARE?

Malware is software designed to harm a computer system or its users. In some cases, malware can exploit software or hardware vulnerabilities to gain access to privileged or private resources and thus compromise the integrity of the system. In other cases, these programs can behave maliciously even without using vulnerabilities (displaying unwanted ads, keyloggers, stealing personal data, etc.). In this case they can harm the user, but do not necessarily affect the integrity of the system.

HOW DOES MALWARE SPREAD?

Even though some malware can be installed without any action from the user, in most cases the installation is done by the user inadvertently. For example, to entice the user to install it, a malicious application can pretend to be a legitimate application and perform malicious actions without the user's knowledge. It is also possible for a legitimate application to become malicious, for example if the application's publisher is hacked or if the publisher unknowingly uses a malicious software component.

WHAT PROTECTIONS EXIST AGAINST MALWARE?

There are generally two types of protection:

- The protections provided by the operating system.
- The protections provided by a pre-installation audit.

Protections provided by the operating system

The operating system is the core software whose role is to provide communication interfaces between applications and hardware and to manage the distribution of available resources between the various applications, but also to ensure the protection of the user and the hardware against malicious software. To do this, it has several tools at its disposal:

- **Application isolation** and the **permissions system**, which restrict how an application can access data from other applications or system functions. To obtain access, the application must request permission from the operating system, which is usually accompanied by a notification to the user who can then accept or refuse this request. The permissions system gives the user important visibility on the application's behavior. For example, it would be abnormal for a flashlight application to request access to the address book.
- Use of **programming interfaces to access system functionality**. In a modern operating system, applications cannot communicate directly with the hardware. They can only access specific features of the hardware, operated by the operating system, and made available via application programming interfaces. Thus, it is the operating system that takes care of the direct communication with the hardware and transmits the results of this communication to the application. This hardware isolation through restricted interfaces considerably reduces the attack surface of malicious software, which can only interact with these interfaces, which only provide access to limited functionalities and offer enhanced security.
- **Detection of malicious programs or behaviors at runtime**, for example by computing an application fingerprint and comparing it to a database of malware fingerprints, **like an antivirus program**. This technique allows malware to be detected and blocked even after it has been installed.

The operating system therefore already provides protection against malware, both from a hardware and user protection perspective. Furthermore, it should be noted **that as long as no vulnerabilities exist within the operating system, it is impossible for an application to compromise the hardware or the operating system itself**. In other words, securing the hardware and the operating system rests exclusively on the operating system developers and never on the application developers.

However, operating system protections are only truly effective if users have an up-to-date system on their devices, which allows them to benefit from security patches, refinements to the permissions system, traceability of application behavior, personal data protection systems, etc.

Protections provided by a pre-installation audit system

Auditing applications prior to user installation **provides additional protection** against fraud or scams, by analyzing and comparing each application description with their actual behaviors. Thus, the auditor acts as a trusted third party for the user to rely on in order to install legitimate applications.

The protections provided by such an audit system are only effective if the audits are thorough. Indeed, a superficial audit that simply compares the description of an application and its visual appearance at launch can only discover obvious malware. An effective audit must be exhaustive and check the application's permissions, the system functions it accesses, the data exchanges with the outside world, the data processing policies outside the application, etc.

Pre-installation auditing can sometimes rely on source code analysis, and ensure that the installed application matches its provided source code, but this situation is rare, and none of the current major operating system developers do source code verification of the third-party applications they audit and distribute in their application stores.

A pre-installation audit can only check the local behavior of an application, and could not, for example, detect the malicious behavior of a developer whose address book application would save the contacts on his server, and then have a malicious use of these personal data stored on his server (dissemination of contacts from the server for example).

HOW TO ENSURE THE SECURITY OF USERS WHILE OPENING UP DISTRIBUTION TO THIRD PARTIES?

The distribution of applications on our smartphones can be done using a more open and decentralized model without jeopardizing the level of user security conferred by practices presented as the state of the art in this field, or even improving it. Indeed, we have seen that the operating system is at the core of malware protection, both against applications that attempt to harm the system or the hardware, but also against fraudulent applications that seek to harm the user. To enhance user security, it is important to take steps at the operating system level:

- Design a **fine-grained permissions system** that allows true traceability of application behavior and limits the attack surface for malware.
- Implement **malicious code detection systems**, which can stop malicious software spread by third party means.
- Implement **digital certificate verification capabilities**, which allow applications to be certified regardless of how they are installed.
- Ensure **tracking and deployment of system updates**, even on older devices, so that all users can benefit from security enhancements or patches on their devices.

We have also seen that **these protections can be complemented by a system of auditing applications** before they are even installed. **At present, these audits are carried out within the application stores that also ensure the distribution of these applications. However, it is easy to decouple these two processes**, thanks to digital certificates, which can both guarantee that an application that one wishes to install corresponds strictly to an application that has been audited and that the description of the application that one wishes to install is strictly identical to the one submitted to the audit process. The operating system can then verify the certificate of an application during its installation or execution. This certificate can also be revoked by the application developer, if they believe that their application has been compromised, by the auditor if they believe that the application should not have been certified, or even blacklisted by the operating system developer, if they believe that the application poses major security issues for the system or the users.

In this framework, the audit can be for instance performed by the teams of the operating system developer, the workload being compensated by an adequate remuneration. Third parties can then offer to distribute the audited and certified applications, including their description.

This audit can also be done directly by third-party application stores, or by auditing organizations independent of the distribution. **The checks made during the audits conducted by current application stores owned by operating system developers are indeed all reproducible by third parties.**

However, it is necessary:

- To supervise application stores or auditing organizations to **meet a standard that guarantees a certain level of review of the applications** offered.
- To ensure a **follow-up over time of the audited applications**, in order to allow the revocation of their digital certificates and, in some cases, the deletion of the application if a problem is discovered after the audit.

Legal deposit : in progress
contact.peren@finances.gouv.fr

PEReN – 120 rue de Bercy, 75572 Paris Cedex 12