Ovaldroid: an OVAL-based Vulnerability Assessment Framework for Android

Martín Barrère, Gaëtan Hurel, Rémi Badonnel, Olivier Festor

Ovaldroid: an OVAL-based Vulnerability Assessment Framework for Android

Martín Barrère, Gaëtan Hurel, Rémi Badonnel and Olivier Festor

INRIA Nancy Grand Est - LORIA, France
Email: {barrere, hurel, badonnel, festor}@inria.fr

***Abstract*—Mobile computing devices and the services offered by them are utilized by millions of users on a daily basis. However, they operate in hostile environments getting exposed to a wide variety of threats. Accordingly, vulnerability management mechanisms are highly required. We present in this demo a novel approach for increasing the security of mobile devices by efficiently detecting vulnerable configurations. In that context, we propose Ovaldroid, an OVAL-based distributed framework for ensuring safe configurations within the Android platform and we present an implementation prototype developed to this end.**

I. Background

The overwhelming technological advances in the broad sense of mobile computing have made end users to experience real computers in their pockets. Android [1], a Linux-based operating system for mobile devices, is nowadays the election of millions of users as the platform for governing their mobile devices [2]. However, despite of the many security improvements that have been done since Android's creation, the underlying operating system as well as services and applications have also evolved providing room for new vulnerabilities [3]. Hostile environments, long patch cycles [4], uncontrolled application markets and the increase of malware[1] [5] worsen the situation. Sensitive data handled by mobile users becomes easily exposed. In that context, managing vulnerabilities is a crucial task that must be addressed in order to ensure safe configurations and to increase the overall security of the system.

In addition, mobile devices usually have limited resources thus optimized lightweight tools should be developed to ensure efficiency without losing functionality. Even though security solutions for Android have been developed by different providers such as [5], there are no current solutions built over solid foundations as well as open and mature standards that foster its adoption and speed up general vulnerability information exchange within the community. In light of this, we propose a novel approach for increasing the security of the Android platform [6], though it could be applied over other mobile platforms as well, using the OVAL[2] language [7] as a means for describing Android vulnerabilities [8]. We put forward a lightweight framework called Ovaldroid and present an implementation prototype that efficiently takes advantage of such knowledge in order to detect and prevent configuration vulnerabilities.

[1]Malicious software including virus, worms and spyware among others
[2]Open Vulnerability and Assessment Language

II. Ovaldroid, an OVAL-based framework for assessing Android vulnerabilities

We have designed the proposed framework illustrated in Fig. 1 as a distributed infrastructure composed of three main building blocks: (1) a knowledge source that provides existing security advisories, (2) Android-based devices running a self-assessment service and (3) a reporting system for storing analysis results and performing further analysis. The overall process is defined as follows. Firstly at step 1, the Android device periodically monitors and queries for new vulnerability descriptions updates. This is achieved by using a web service provided by the security advisory provider. At step 2, the provider examines its database and sends back new found entries. The updater tool running inside the Android device synchronizes then its security advisories. When new information is available or configuration changes occur within the system, a self-assessment service is launched in order to analyze the device at step 3. At step 4, the report containing the collected data and the results of the analyzed vulnerabilities is sent to a reporting system by means of a web service request. At step 5, the obtained results are stored where they can be analyzed with different purposes such as forensic activities or statistical analysis.

In order to provide a computable infrastructure to the proposed approach, a running software component inside Android capable of performing self-assessment activities is required. Our implementation prototype has been developed to be compliant with Android platforms starting at version 2.3.3, thus covering almost 80% of the Android market share. It is composed of four main components: (1) an update system that keeps the internal database up-to-date, (2) a vulnerability management system in charge of orchestrating the assessment activities when required, (3) an OVAL interpreter for the Android platform and (4) a reporting system that stores the analysis results internally and sends them to an external reporting system. Fig. 2 depicts the main operational steps performed during the self-assessment activity and the connection with the mentioned four main components. The prototype is executed as a lightweight service that is running on background and that can be awakened by two potential reasons as shown at step 1. The updater listener listens the vulnerability database updater component and will be notified when new vulnerability definitions become available.

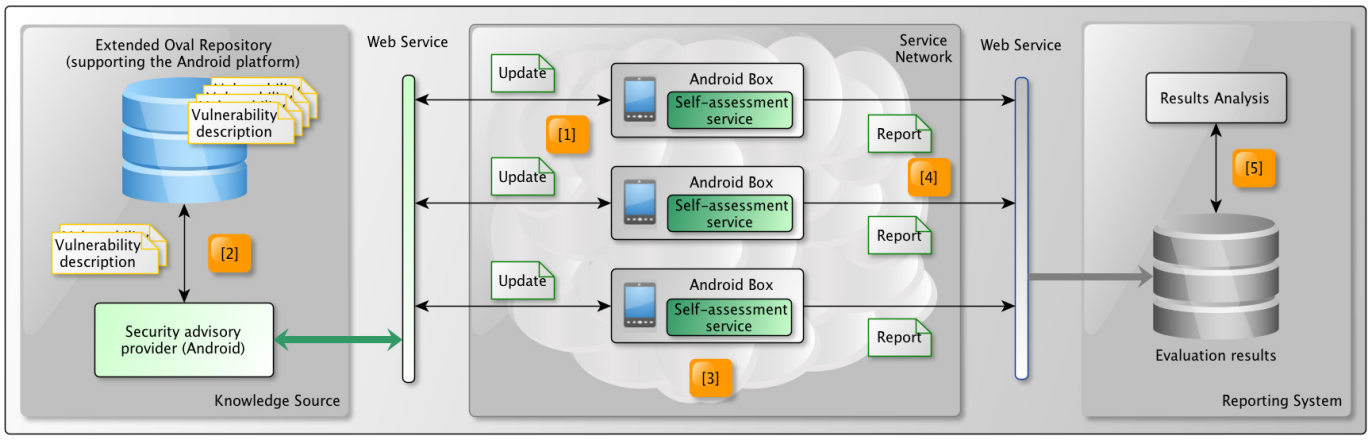

Fig. 1: Ovaldroid, an OVAL-based vulnerability assessment framework for the Android platform

The event bus listener uses the Android broadcast bus to capture notifications about system changes. If new vulnerability definitions are available or system changes have been detected, a vulnerability definition selection process is launched at step 2. This process is in charge of analyzing the cause that has triggered the self-assessment activity and determining which assessment tasks must be performed. At step 3, the vulnerability manager component uses the services of XOvaldi4Android in order to perform the corresponding assessment activity. At step 4, the results of the assessment are stored in the internal results database and sent to the external reporting system by performing a web service request. Finally, a local notification is displayed to the user if new vulnerabilities have been found in the system. XOvaldi4Android plays a fundamental role within the proposed framework because it is in charge of actually assessing the Android system. XOvaldi4Android is an extension of XOvaldi [9], a multi-platform and extensible OVAL interpreter. We have ported the XOvaldi system to the Android platform obtaining a 94 KB size library.

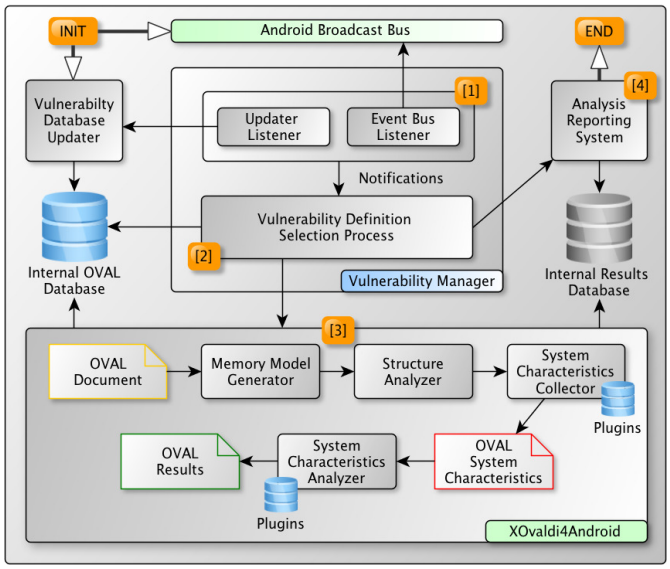

Fig. 2: Self-assessment service high-level operation

III. The demo

In this demo we will show how we can increase the overall security of the Android platform by automatically integrating and assessing vulnerability descriptions over real Android devices in a distributed manner. To do so, we will consider OVAL descriptions of real vulnerabilities stored in our knowledge source that will be automatically consumed and analyzed by running Android devices. During the presentation, a visual medium illustrating the key concepts of our framework Ovaldroid will be provided. In addition, scenarios where new security advisories become available will be presented as well as the impact of Ovaldroid over those devices under control. We will also analyze the obtained results that support the feasibility of our solution. To conclude, the integration of vulnerability management activities into mobile environments poses hard challenges. Supporting vulnerability awareness constitutes the first step towards secure self-managed infrastructures capable of detecting and remediating potential security breaches.

Acknowledgements

This work was partially supported by the EU FP7 Univerself Project and the FI-WARE PPP.

References

[1] "Android." http://www.android.com/. Last visited on January, 2013.
[2] "Gartner." http://www.gartner.com. Last visited on August, 2012.
[3] W. Enck, D. Octeau, P. McDaniel, and S. Chaudhuri, "A Study of Android Application Security," in *Proceedings of the 20th USENIX Conference on Security*, SEC'11, (Berkeley, CA, USA), USENIX Association, 2011.
[4] T. Vidas, D. Votipka, and N. Christin, "All Your Droid Are Belong To Us: A Survey of Current Android Attacks," in *Proceedings of the 5th USENIX Conference on Offensive Technologies (WOOT'11)*, (Berkeley, CA, USA), pp. 10–10, USENIX Association, 2011.
[5] "Lookout Mobile Security." https://www.mylookout.com/mobile-threat-report. Last visited on January, 2013.
[6] M. Barrère, G. Hurel, R. Badonnel, and O. Festor, "Increasing Android Security using a Lightweight OVAL-based Vulnerability Assessment Framework," *In Proceedings of the 5th IEEE Symposium on Configuration Analytics and Automation (SafeConfig'12)*, Oct. 2012.
[7] "The OVAL Language." http://oval.mitre.org/. Last visited on January, 2013.
[8] "OVAL Language Sandbox." http://oval.mitre.org/language/sandbox.html. Last visited on January, 2013.
[9] M. Barrère, G. Betarte, and M. Rodríguez, "Towards Machine-assisted Formal Procedures for the Collection of Digital Evidence," in *Proceedings of the 9th Annual International Conference on Privacy, Security and Trust (PST'11)*, pp. 32 –35, July 2011.