IMA/EVM Study for Automotive Grade Linux

Objectives

The objectives of this study are : - Study IMA/EVM features - Test it on AGL Platform - State on the usage in production

IMA - Integrity Measurement Architecture

IMA is a linux feature use to control integrity of defined files. It use the extended file properties to store checksums used to validate integrity. It consist of three main parts.

- IMA-measurement : The measurement part maintains a runtime measurement (checksums) list of the file we want to follow. This file can't be modified at runtime. This list can (and should) be anchored to a Trusted Platform Module (external hardware component) to ensure that it can't be corrupted.
- IMA-appraisal : The appraisal part maintains a similar list, but use it to ensure integrity offline. Actually it checks defined files at boot time. Otherwise it works quite the same.
- IMA-audit : This part quite simply log the interactions with the two upper parts.

EVM - Extended Verification Module

EVM is the security counterpart of IMA. It's used to sign the hash used to control integrity. It can be configured in many ways, mostly the keys that can be used, and their origins. It should also be anchored on a TPM

IMA Activation

To activate IMA, you have first to build your linux kernel with the feature. Then to enable it, you have to set at boot time (via boot parameters) the type of checking you'll be doing. In addition, to add a more precisely tuned control, you can you a custom policy file (located in /sys/kernel/security/ima/policy)

More information about it in : - `https://sourceforge.net/p/linux-ima/wiki/Home/` - `https://wiki.strongswan.org/projects/strongswan/wiki/IMA`

IMA Usage

To use IMA, you can either use some of the developped but experimental tools (ima*measure) or simply by checking the files* "*violations*" *and* "*ASCII*runtime*measurements*" (located in /sys/kernel/security/ima/).

As for the activation, more information about it in : - `https://sourceforge.net/p/linux-ima/wiki/Home/` - `https://wiki.strongswan.org/projects/strongswan/wiki/IMA`

Integration and Testing on AGL

We encountered many problems with the integration on AGL, mainly linked to difficulty to manipulate the feature, and building the AGL distro with it, with the Yocto project. We can however already present some of our results from the documentation research.

IMA's problems

We don't recommend IMA's and EVM's use as part of the AGL project, for there is many issues with it. - On the subject of security, the feature doesn't protect directories. It means that if someone can access and modify files, by using untrusted directories and links (sym/hard) the integrity and signing feature can be bypass. - On the subject of safety, on embedded systems, power outage are a risk frequent enough to have to take it in account. However, IMA doesn't do it. A power outage while modifying hashes might generate alerts at best , and brick the system at worst. - On the subject of production deployment, as IMA is next to useless without EVM's signing, the keys storage is a problem to tackle. However, generate and manage TPMs for multiple architecures and the key generation make it very difficult to use without having very specific hardware setups. - Finally, for the reasons above, and for maturity reasons, the features are explicitly marked as depecated in production, and to use only in developpement environement.

Conclusion on IMA/EVM

To conclude, IMA should be a great tool to check the integrity of the files of our system while offline (mainly), but concretely, it's not yet mature enough to be exploited in production, even more on embedded project such as AGL.

Source of informations

- https://code.woboq.org/linux/linux/security/integrity/ima/ima_main.c.html
- https://sourceforge.net/p/linux-ima/wiki/Home/
- https://wiki.strongswan.org/projects/strongswan/wiki/IMA
- https://lwn.net/Articles/137306/
- https://lwn.net/Articles/733431/
- https://lwn.net/Articles/137311/

Special thanks

- IotBzh
- Patrick Ohly