INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC 1/SC 29/WG 11
CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JTC 1/SC 29/WG 11
MPEG2017/m39944
January 2017, Geneva, Switzerland

Source: **DMAG-UPC**
Status: **Proposal**
Title: **GENIFF (GENomic Information File Format) v2: Security and signature issues**
Authors: **Daniel Naro, Jaime Delgado, Silvia Llorente (Distributed Multimedia Applications Group – Universitat Politècnica de Catalunya)**

Contents

1 Introduction 2
2 Need for integrity and authenticity verification 2
3 Thoughts on implementations 2
3.1 Algorithms 2
3.2 Formatting signatures 2
3.3 Certification authorities 3
4 Current implementation 3
5 Acknowledgements 3
6 References 3

1 Introduction

This document describes early experiments on adding electronic signatures to the GENIFF (GENomic Information File Format) [1].

The privacy rules, specified with XACML in GENIFF, allow us to define who is granted access, and under which circumstances, to user-defined regions of the genomic file. With the help of encryption, we ensure that, even if the rules are circumvented, only those users with the key will be able to access the data.

Depending on the encryption option, it might however be possible to modify the file in order to get access granted. For example, if the user prefers not to encrypt the file or to encrypt it only with one key across the whole file, it could be possible to change the regions defining the privacy rules and obtain the desired (and non-authorized) access.

2 Need for integrity and authenticity verification

The described issue is addressed with the definition of signatures. Providing a pair of keys (public and private), users will be able to sign portions of the content, proving they were not tampered with.

The regions containing the header information for each level in the hierarchy are of main interest. These portions are meant to remain in plaintext, and are therefore easy to modify. The genomic payload should also be signed, but its very large size makes the verification of the authenticity of the entire file a challenge (if meant to check it almost instantly).

3 Thoughts on implementations

3.1 Algorithms

Cryptography provides us with different algorithms to perform the signature, such as RSA or DSA. Such algorithms might not be enough in the case where the foreseen time of life of the genomic file might be so long that quantum computing could be a real threat to the authentication mechanism. To this end, the format should support multiple signature methods, some belonging to post-quantum cryptography. The format should also consider the need to include multiple signatures of the same content, as a fail-safe solution if in the future one of the methods is to be broken.

3.2 Formatting signatures

In the case of including just one signature obtained with an algorithm fixed by the standard, it might be enough to include only the value of the signature (e.g. the 2048 bits of an RSA signature). However, as mentioned in the previous section, and due to the lifetime of the files, fail-safe mechanisms in the form of multiple signatures might be required. To this end, defining regions of the file containing multiple XML signatures could help to structure the format, as they allow to indicate the methods used in the authentication process.

In the case of the genomic payload, and depending on the chosen granularity for the authentication, including multiple signatures might be too much overhead. This is even more likely if the signature formatting introduces an overhead of its own. In that case, another format for the authentication might be required, maybe stepping back to a plain RSA signature. In doing so, we might be losing options in security. Considering the compression as a hurdle for file tampering might only apply in the case where the original data is not known. Otherwise, compressing the modified payload and basing the attack upon the observed difference in the deflated payload could be enough.

3.3 Certification authorities

The transport of the public key to be used in the description (and possibly the multiple keys if more than one algorithm is used) will either require an external channel or to be included in the file format. In either case, the authenticity of the key is likely to be confirmed by a chain of trust, which would have to be defined. In the case of working with anonymized content, fields such as the name in an X509 certificate should be repurposed to transport the sample identifier, for example.

4 Current implementation

The implementation currently provided introduces at the end of each header a field of 2048 bits containing the RSA signature. These signatures protect against non-authorized modification of the header's attributes and the header's children.

For the moment, only one private key is used for the entire file. In the future, the introduction of a proper signature format, such as the previously introduced XML signature, will simplify the usage of different signature methods and different public keys (for example one per individual).

5 Acknowledgements

The work done in this proposal has been partially supported by the Spanish Government under the project: Secure Genomic Information Compression (GenCom, TEC2015-67774-C2-1-R).

6 References

[1] Jaime Delgado, Silvia Llorente, Daniel Naro et al., ISO/IEC JTC 1/SC 29/WG 11 M39940, GENIFF v2, December 2016.