THE USE OF CRYPTOGRAPHIC PRINCIPLES WITHIN METROLOGY SOFTWARE

S. Parkinson, A .P. Longstaff, A. Crampton, S. Fletcher, G. Allen, A. Myers
Centre for Precision Technologies, University of Huddersfield,
Queensgate, Huddersfield, West Yorkshire, HD1 3DH, UK

Introduction

To enable a thorough software design, a rigorous process of establishing the desired requirements must be performed. This methodology is certainly used for the production of metrology software. When establishing the requirements, both metrology and security aspects will be considered to allow for design and production of robust, secure and functional software. This will typically result in effort being spent to satisfy both metrology and security requirements separately. An illustration of the current effort is shown in Figure 1A. This poster presents a fundamental shift in philosophy to the design and production of metrology software, which is illustrated in Figure 1B.

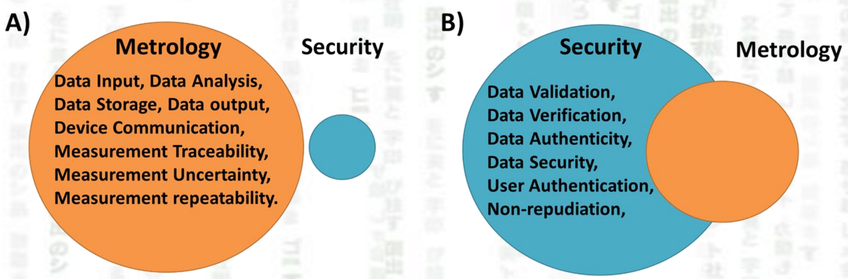

Figure 1. Showing the design effort of the current and proposed methodology. The effort is represented by the size of the circle.

Requirements

There is a common set of metrology related functional requirements [1], with differences regarding the connectivity and distribution of the software. These basic requirements are: (1) Traceability, (2) Uncertainty, and (3) Repeatability.

The security requirements of an internet-enabled metrology system are well explored [2]. However, the security requirements for software that makes use of a different networking medium, or operates in a standalone manner, should be given the same emphasis to ensure that adequate security precautions are always taken. The main software security requirements are: (1) Data security, (2) Data integrity, (3) Data authentication, and (4) User authentication.

Security-centric Design

The new philosophy proposed in this poster is a security-centric approach to the design and production of metrology software. As illustrated in Figure 2, by carrying out thorough design and implementation of the security requirement, the functionality requirements can also be satisfied. This strong relationship between the security and metrology requirements is underpinned by the captured metrology data. For example, information making the performed measurement repeatable and traceable, which includes all processing comparisons and uncertainties, should be recorded within the data. This means that employing good security procedures will maintain the integrity of the metrology data, thus preserving its qualities.

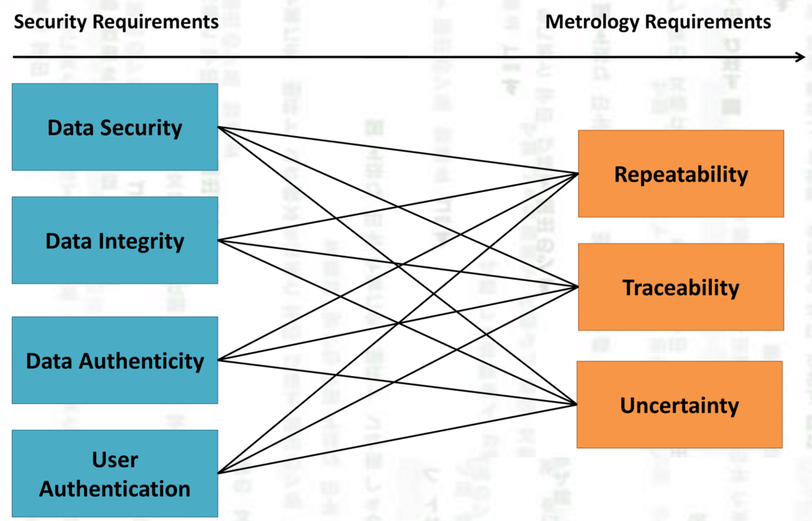

Figure 2. Illustration showing the relationship between the security and the metrology requirements of a software-based metrology system

References

1. BS, Measurement management systems - Requirements for measurement processes and measuring equipment. 2003, BSI: London, United Kingdom.
2. R. M. Barker., Software Support for Metrology Best Practice Guide No. 19, in Internet-enabled Metrology Systems. 2006, National Physical Laboratory
3. Å. Sand, H. Slinde, T. A. Fjeldly, A Secure Approach to Distributed Internet-Enabled Metrology. IEEE Transactions on Instrumentation and Measurement, 2007. **56**(5): p. 1979 - 1985

Cryptographic Principles

Previous efforts have incorporated the use of cryptographic functionality in a bottom up approach from design to implementation [2, 3]. Here we propose a similar procedure, however, now we are concentrating on satisfying the security requirements first in the knowledge that by doing so we are also satisfying the metrology requirements. The asymmetric public-key infrastructure provides a function set that can satisfy all the security requirements of metrology software. The following list states the cryptographic functionality that can be implemented to meet the security requirements, and Figure 3 shows the many locations where a software engineer might consider their implementation.

1. Data security - Asymmetric encryption/decryption
2. Date integrity - One way hash function
3. Data authenticity - Digital signature
4. User authentication - Digital signature

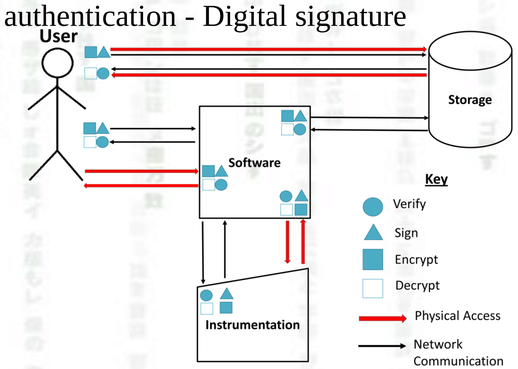

Figure 3. Illustration of where to insert the cryptographic functionality within a software system

Conclusion

This poster demonstrates the relationship between the security and metrology software requirements. A security-centric philosophy to the design and implementation of metrology software is also presented. Following this philosophy, and the use of cryptographic functionality within an object-oriented language, can save on programming effort to achieve secure, reliable and functional software. However, further work will be performed to validate the feasibility of the presented philosophy for both small and large scale software development projects.