MANAGEMENT OF CRITICAL MACHINE SETTINGS FOR ACCELERATORS AT CERN

W. Sliwinski, P. Charrue, V. Kain, G. Kruk, CERN, Geneva, Switzerland

Abstract

In high energy and high intensity accelerators as the LHC, the energy stored in the beams is orders of magnitude above the damage level of accelerator components like magnets. Uncontrolled release of this energy can lead to serious damage of equipment and long machine downtimes. In order to cope with these potential risks Protection Systems were developed at CERN including two software systems: MCS (Management of Critical Settings) and RBAC (Role Based Access Control). RBAC provides an authentication and authorization facility for access to the critical parts of the control system. A second layer of security is provided by MCS which ensures that critical parameters are coherent within the software and hardware components and can only be changed by an authorized person. The MCS system is aimed at the most critical parameters in either potentially dangerous equipment or protection devices (e.g. Beam Loss Monitors). It is complementary to the RBAC infrastructure. Both systems are fully integrated in the control system for the LHC and SPS and were successfully commissioned already before first beam in the LHC. This paper will describe the MCS architecture, current status and its operational deployment in the LHC.

INTRODUCTION

For certain equipment systems, the interlock settings may be resident in the hardware and cannot be changed remotely. Nevertheless, many systems have to provide configurable interlock settings as they have to be calibrated or defined with beam during beam operation. In order to address these special needs, a specific solution was introduced: **M**anagement of **C**ritical **S**ettings (MCS) [1] which is nowadays an integral part of the control system for the LHC and SPS. MCS interfaces the repository of operational settings and implements a secure channel for changes of the interlock settings.

MCS handles only the key machine protection related interlock settings, which should be modified infrequently, during initial commissioning, setting-up, recovery from interventions or after machine stops. It is not used for managing normal operational settings; it is aimed at the update of critical interlock settings which are normally considered as “almost locked”, providing maximum security against accidental or uncontrolled modification and strictly limiting the access to these critical settings.

MCS ARCHITECTURE

Overall Architecture

The MCS infrastructure was integrated as part of the core of the control system for the LHC and SPS. The overall architecture including the data flow accompanying every modification of a critical property is illustrated in Figure 1.

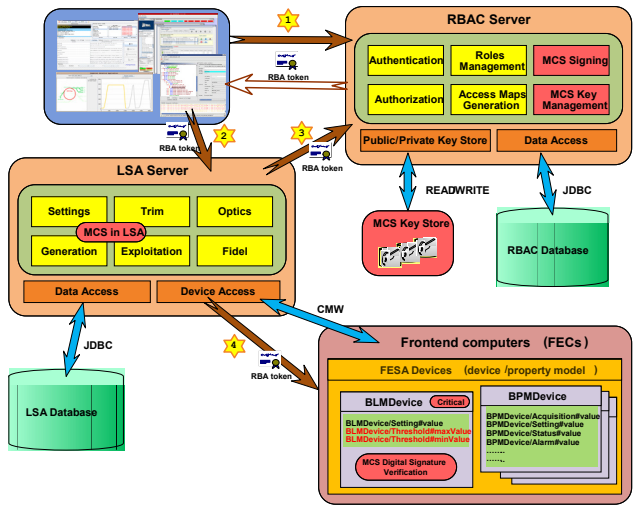

Figure 1: The overall MCS architecture.

In the presentation tier, client applications running in the CERN Control Centre (CCC) trigger the modification of selected critical settings. This request is handled in the middle tier by the high-level control system called LHC Software Architecture (LSA) [5,6]. LSA is using the security services provided by the Role Based Access Control (RBAC) [2,3,4] system. In the lower tier, MCS integrates with the Front-End Software Architecture (FESA) [8] dedicated to the real-time control of the accelerator hardware.

In the following paragraphs, introduced control sub-systems are presented together with the detailed description of their integration within the MCS infrastructure.

Role Based Access Control (RBAC)

Critical settings can compromise the safety of an accelerator, thus it is important to ensure that they can only be changed by an authorized equipment expert or trained personnel and not through uncontrolled access to the front-end machines. Moreover each expert should have the right to modify only a subset of the parameters within his or her domain of expertise. “Super users” cannot be allowed. These requirements were achieved by incorporating the authentication and authorization services provided by RBAC within MCS.

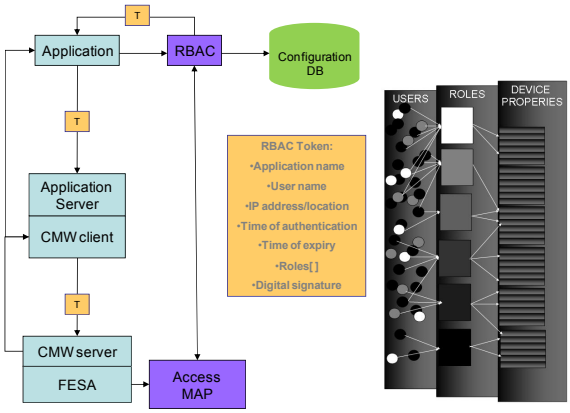

Figure 2: RBAC in the control system.

The security scheme in RBAC (see Fig. 2) is based on a concept of roles which are given to selected users and assigning permissions to roles (Access Rules) to access certain hardware properties. RBAC facilitates the authentication process by providing software login components for both Graphical User Interface (GUI) and non-GUI clients. Once the user is authenticated (Fig. 1, operation 1), a security credential called "RBA token" is generated which represents the user's identity in the authorization process used with predefined Access Rules.

In addition to RBAC's authentication and authorization MCS provides a mechanism to guarantee data integrity at all times using the digital signatures. RBAC was extended to provide the management of the public/private key pairs (stored in the private, local Key Store) as part of the role management. The special MCS roles are associated with the public/private key pairs. The private keys are kept secret and never leave the RBAC server. The digital signatures are generated with the private key whenever LSA sends critical property data to the RBAC server with a request for signing, given that the client-side user has the required MCS role, according to the Access Rule for the critical property. The public keys are made available to front-ends, namely to FESA servers and other software processes to verify the MCS signature (see Fig. 1, operation 4) when critical properties are to be updated.

LHC Software Architecture (LSA)

LSA is the core, 3-tier, high level controls framework for the LHC, SPS with its transfer lines and the LEIR accelerator. It provides advanced management of operational machine settings and covers all of the most important aspects of the accelerator controls: the optics configuration, the operational parameters space, the settings generation, the settings update (trim) and operational exploitation, hardware exploitation and beam based measurements. Moreover, it exposes a generic client API (Application Programming Interface) to all of its core functionality, which is used by the operational applications.

Management of Critical Settings was introduced into the LSA framework as an extension to the existing system; particularly the Settings and Trim modules were adapted to the MCS requirements. The LSA database [7] is the master source for all the operational settings, including the critical ones. Moreover, information about a property being "Critical" must be registered in the LSA database which is facilitated through the LSA Parameter Configuration application. To fulfil the MCS requirements the LSA settings database model was extended in the two following areas:

- An additional possible attribute for a property and related parameters: Critical
- An additional data (setting) for a critical parameter, namely the digital signature computed for all the fields of a critical property. It is stored with the data.

MCS within LSA can manage (sign and verify) any of the data types below:

- Scalar values (integer, short, float, double, string)
- Arrays of scalars
- Two-dimensional arrays of scalars

Any operation using LSA, including a change of a critical parameter, must be followed by creating and storing a new signature. Thus any of these typical LSA operations, providing the MCS mechanism below, require RBAC authentication and authorisation:

- Generation of new settings
- Trim (update) & Copy of an existing setting
- Download of current settings to the equipment
- Acquisition of hardware values into new, initial critical settings

Once a modification (e.g. LSA trim or generation operation) of a critical setting is requested (see Figure 1, operation 2) the new settings values are calculated by the Trim module. The LSA Trim module which is used in all settings modifications deals with the additional requirements for the critical settings. All the "fields" of possibly complex critical properties have to be combined into a single message. This message is then sent to the RBAC server with the corresponding role as key identifier to get the signature back. The updated setting together with the signature is kept in the LSA database. Any person (not requiring the MCS role) can send the critical setting together with the signature to the front-end at any time afterwards. Only the data content signed with the role of the authorized person will be accepted on the front-end as discussed in the next section.

Front-End Software Architecture (FESA)

In the lower tier of the front-end computers (FECs), performing real-time control of the accelerator equipment, the FESA framework is deployed. FESA is an important component of the MCS infrastructure - the digital signature verification scheme is therefore an integral part of the FESA framework. After marking a property as Critical in the LSA database, an XML configuration file, containing the critical property's name together with the corresponding public key is generated (using the LSA Parameter Configuration application) and made available to the front-end. The non-existence of this file, a partial bypass of MCS, can be detected at any time with LSA tools. With these XML files, FESA knows that it is

dealing with critical properties and expects digital signatures for the properties defined in the configuration file, for each "set" command. If the signature is missing or the public key verification according to Fig. 3 fails, the setting is rejected. A special effort had to be made to end up with the same digest functions on the signing side and front-end side for different platforms.

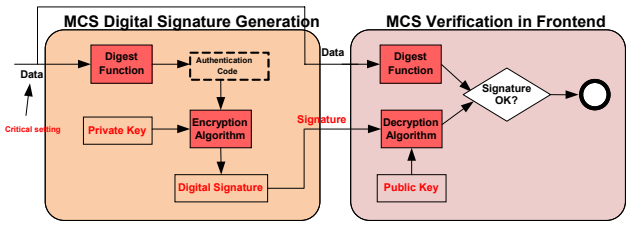

Figure 3: MCS signature verification in FESA. The public key decrypts the received signature. The resulting digest function is compared with the digest function computed from the data on the front-end. If the comparison is successful, the critical property is updated on the front-end.

INTEGRITY OF CRITICAL SETTINGS

The XML files could be deleted on the front-ends, data on the front-ends can become corrupted due to for example radiation. As we are talking about machine protection relevant settings, in case of critical settings data integrity is of a great importance. Consistency checks were introduced to ensure the data integrity. These checks were implemented in the LSA Exploitation module. The "true" source is the LSA database, guaranteed by storing the signature with the data. Three tests are available:

- Database check – checks consistency of the db critical settings with the db signature (uses public key signature verification algorithm)
- Online check – checks consistency of the critical settings between the LSA database and the hardware
- Frontend deployment check – checks if the MCS configuration file for a frontend exists and has the correct contents

The MCS consistency checks can be triggered either on demand from the GUI application LSA Parameter Configuration or programmatically using the high level LSA API which exposes the different MCS checks as generic commands. The latter approach is already integrated with the LHC Sequencer [9], which was programmed to run the MCS checks before and during each LHC fill operation. Furthermore, the Software Interlock System [10] was also configured to execute the MCS checks for critical equipment systems during beam operation.

MCS DEPLOYMENT IN LHC

The safe running of the LHC will be partly due to highly reliable protection systems such as the Beam Loss Monitor System (4000 monitors with energy dependent thresholds), the Collimators (optics and beam position dependent collimator jaw interlock thresholds), the LHC Beam Dump System, the Beam Interlock System with its Safe Machine Parameters (e.g. probe intensity flag threshold), interlocked Beam Position Monitors and many more. All BLM thresholds, all collimator thresholds, the modifiable Safe Machine Parameters thresholds etc. are all managed by MCS. For the time being it is covering about 15 different hardware systems with some of them using 1000s of thresholds.

CONCLUSIONS

The Management of Critical Settings (MCS) is an integral part of the control system for the LHC and SPS, managing machine safety critical hardware and virtual parameters. It is using a state-of-the art Role Based Authentication and Authorization service together with Public/Private Key digital signatures to ensure data integrity at all times. MCS has been designed to protect the nominal path of data access with the aim to avoid operational errors and consequently long down times. It has been in use since 2007 and more and more systems in the LHC and also other accelerators at CERN make use of settings protected by MCS.

REFERENCES

[1] V.Kain et al., "Management of Critical Settings and Parameters for LHC Machine Protection Equipment", CERN-LHC-CI-ES-0003, 2006, CERN, Geneva, Switzerland.
[2] S.R.Gysin et al., "Role-Based Access Control for the Accelerator Control System at CERN", ICALEPCS'07, Knoxville, Tennessee, U.S.A.
[3] K.Kostro et al., "Role-Based Authorization in Equipment Access at CERN", ICALEPCS'07, Knoxville, Tennessee, U.S.A.
[4] A.D.Petrov et al., "User Authentication for Role-Based Access Control", ICALEPCS'07, Knoxville, Tennessee, U.S.A.
[5] M.Lamont et al., "LHC Era Core Control Application Software", ICALEPCS'05, Geneva, Switzerland.
[6] G.Kruk et al., "LHC Software Architecture (LSA) – Evolution toward LHC beam commissioning", ICALEPCS'07, Knoxville, Tennessee, U.S.A.
[7] C.Roderick et al., "The LSA Database to Drive the Accelerator Settings", ICALEPCS'09, Kobe, Japan.
[8] M.Arruat et al., "Front-End Software Architecture", ICALEPCS'07, Knoxville, Tennessee, U.S.A.
[9] V.Baggiolini et al., "A Sequencer For the LHC Era", ICALEPCS'09, Kobe, Japan.
[10] J.Wozniak et al., "Software Interlocks System", ICALEPCS'07, Knoxville, Tennessee, U.S.A.