# AVOIDING SECURITY AND PRIVACY ISSUES BY USING SPECIFIC PATTERNS IN BIOMETRIC AND RFID SYSTEMS

**Valentin Corneliu PAU, Professor PhD,**

*Titu Maiorescu University, Computer Science Department*

**Marius Iulian MIHAILESCU, Assistent Professor, PhD candidate,**

*Titu Maiorescu University, Computer Science Department*

**Octavian STANESCU, Inf.**

*Titu Maiorescu University, Computer Science Department*

**v_pau@utm.ro, mihmariusiulian@gmail.com, octavian_stan@yahoo.com**

**Summary:** In this paper we will cover the most important aspects of design patterns and their major role in developing biometric and RFID applications. RFID is one of the most important technologies for the future of applications. Another important aspect is providing solutions for different types of threats and attacks which take place in a biometric or RFID system

**Keywords:** RFID, biometric, ATHOS, holographic signature, pixel

## 1. Introduction

Numerous issues unresolved regarding the detailed technical and sheer operational capabilities of RFID technology are taken into consideration in this article. Because of the large number of considerations which must be undertaken, only few intangible and theoretical considerations, such as privacy, social, security, ethical and architectural are touched in this paper and lead us to the problems that biometrics and RFID technology have raised.

When we wish to apply biometrics and RFID in production, it needs to be integrated into the used IT systems. Untill today system designers cannot rely on a standard template as a solution for integrating RFID technology in the manufacture software systems.

There two categories of patterns for RFID and biometric systems presented here: architectural patterns which are for distributing context data and architectural patterns for distributing filter logic.

The first category provide solutions for managing the flow of context data along the production process. The term ‘context data’ refers to workflow data for controlling the process and security solutions are presented also. The following patterns (RFID Pipeline Pattern, Pulled Context Pattern and Pushed Context Pattern) adress the management of stored formation in the automation pattern for manufacturing. These patterns will provide solutions to the distribution of workflow data, state information, and configurations along production steps. Security aspects and configurations will be presented.

The second category is referring to architectural patterns for distributing filter logic. Before business applications can be used there are several filtering and preprocessing steps for the raw RFID which is read. IT system had to transform the stream of tag observations in business relevant information. As we can see in [1], we name this transformation of raw events into primitive events and subsequently into more complex events.

## 2. RFID-pipeline pattern

The pattern, named RFID-pipeline, defines a logic process for deployment which helps triggering the production steps. The strongest points of this pattern are reliability and scalability of the system.

Figure 1 shows the data flow and physical deployment defined by the RFID-pipeline pattern. Activities in an RFID pipeline fall into three sequential parts. (1) Initializing the pipeline, (2) sequentially triggering and executing production steps, and (3) finalizing the pipeline. In the first step, initial state information, workflow data, and configurations are written on an RFID tag. In most cases the initialization would be triggered in the back-end system because it holds the required information. It follows the sequence of triggering and executing production steps. The processing-logic for each step is deployed on an edge PC on the plant floor. Each step reads state information from the previous step, workflow data, and configurations from the RFID tag. [14]

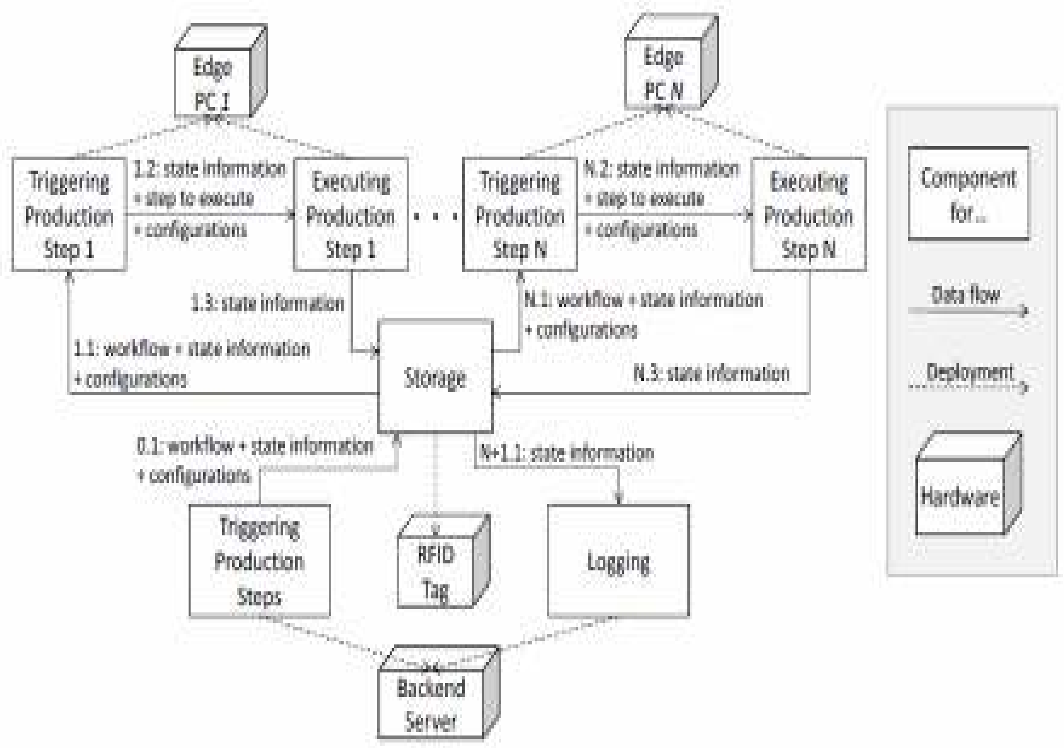
Edge PC 1
Edge PC N
Triggering Production Step 1
1.2: state information + step to execute + configurations
Executing Production Step 1
Triggering Production Step N
N.2: state information + step to execute + configurations
Executing Production Step N
1.3: state information
N.1: workflow + state information + configurations
1.1: workflow + state information + configurations
Storage
N.3: state information
0.1: workflow + state information + configurations
N+1.1: state information
Triggering Production Steps
RFID Tag
Logging
Backend Server
Component for...
Data flow
Deployment
Hardware

Fig. 1. The RFID-pipeline pattern[14]

### 3. Pulled-Context Pattern

The strentgts of this pattern consist in the fact that he support the scalability of the systems. It make the process of access very easy at the production stations and he simplifies the control of data flow.

Figure 2 shows the data flow and physical deployment that the pulled-context pattern defines. The data flow starts with reading the identifier from the RFID tag at a production station. The edge PC at the production station uses this identifier to pull the required context information for the production from a central coordination component in the back end. It is important to note that distance reads of tags RFID can help in conducting the pulling step before the data is actually required. The central coordination component sends context data for the production to the corresponding edge PC as response to the pull request. After conducting the production step, the edge PC transfers updated status information back to the central coordination component. This procedure is repeated afterward for each production step. [8, 14]

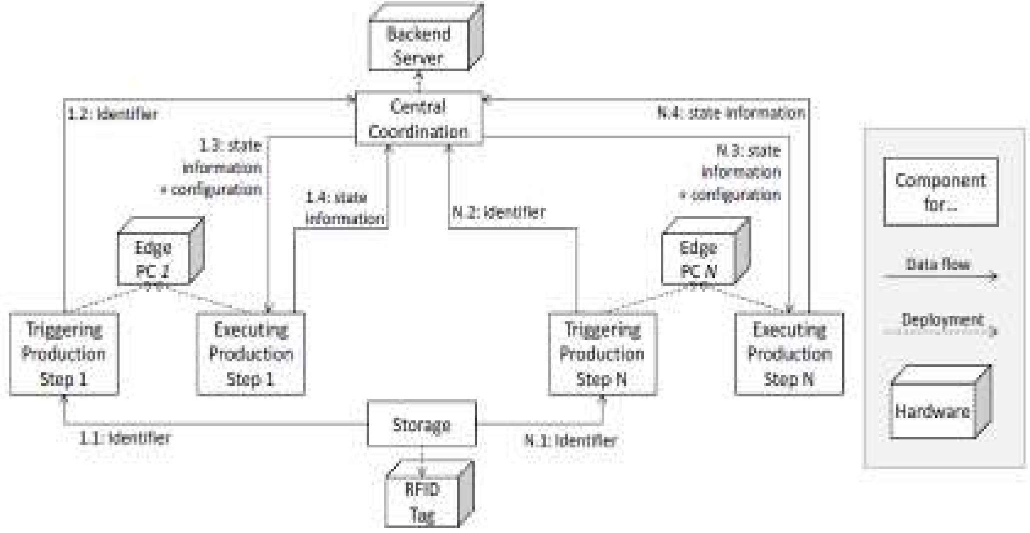

Fig. 2. The pulled-context pattern [8,4,14]

## 4. Thin-Filter Edge Pattern

The thin-filter edge pattern describes the deployment of preprocessing of RFID data on different hardware tiers. The pattern represent an architectural alternative to thickfilter edge pattern. The strength of the pattern consist in simplicity. It fits best to infrastructures with a thin edge tier and to applications where RFID is used for process monitoring.

The pattern directs raw RFID events through a pipeline of filtering and preprocessing operations, see Figure 3 right. The first operations in the pipeline run on edge PCs that control the RFID readers. (Note that some readers have powerful computation capabilities and can replace the edge PC in this pattern.) Subsequent filter operations run in the back end before data is passed on to higher level applications. [14]

The most important advantage of the thin-filter edge pattern is that it keeps processing at edge devices simple. It matches with the typical distribution of filter operations in RFID middleware solutions. The presented solution are focus on monitoring applications e.g., for logistic tracking or warehouse management. For manufacturing this pattern is most suitable if RFID is used for monitoring and documentations rather than for real-time control of production steps. [14,6,8]

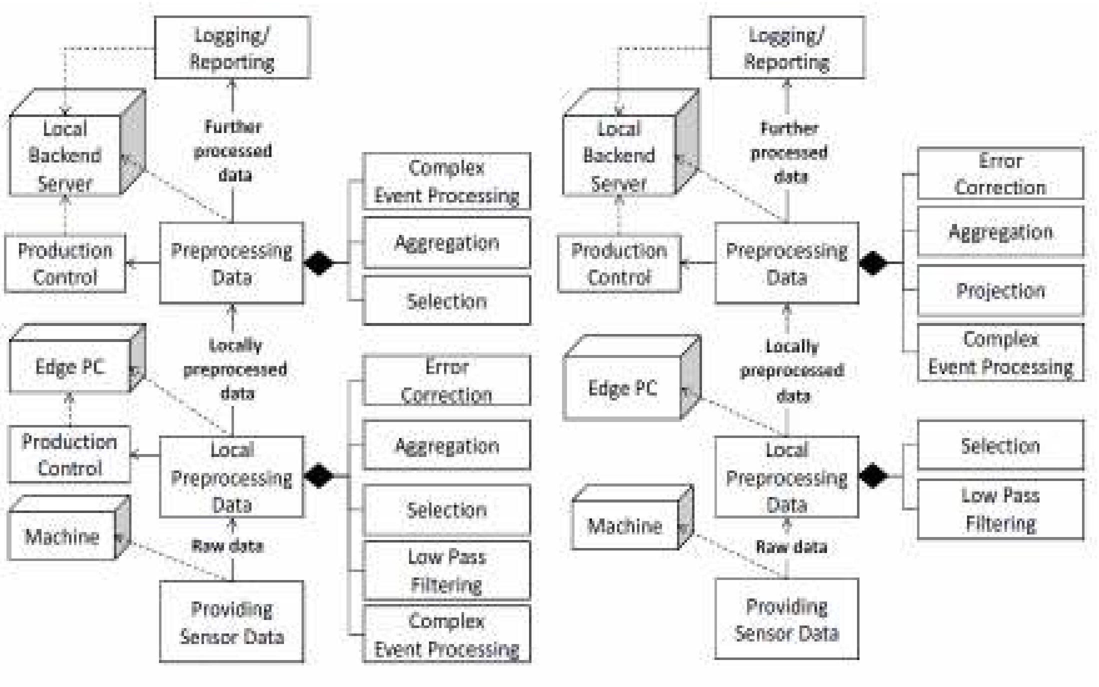

Fig. 3 – Thin-Filter Edge Pattern

## 5. Conclusions and Results

We have study a set of patterns for the integration of RFID in business application which was mainly studied in the field of logistic application. We have taked into consideration the EPVglobal developed standards for capturing and exchanging RFID events. These standards defines interfaces for tracking and tracing applications. We presented architectural patterns which provide concrete design option for integrating RFID in manufacturing IT systems.

We have combine cryptography and biometrics into the patterns that we presented to ensure privacy and security of the data. Finnaly we suggest an actual implementation of the patterns proposed on real biometrics data. The resulting error rate is acceptable and it is not worse that the best error rate of a single characteristic of biometric systems on which is based. Our study open a large way for a scale applicability of privacy for biometric systems. Table 1 show an overview of how the patterns can be combined.

|  | RFID Pipeline | Pulled Context | Pushed Context | Thin Filter Edge | Thick Filter Edge | One-Tier Control | Multi-Tier Control |
|---|---|---|---|---|---|---|---|
| RFID Pipeline |  | - - | - - | + | + + | - | + + |
| Pulled Context |  |  | - - | + | + + | - | + + |
| Pushed Context |  |  |  | + | + + | - | + + |

*Table 1. Combining the presented architectural patterns*

## 6. References

[1] Ivantysynova L. Klafft M., Ziekow J., Gunther O., and Sekin K., ''RFID in Manufacturing: the Investment Decision", In Proceedings of the Pacific Asia Conference on Information Systems, India, 2009.

[2] Sabbaghi A., Vaidyanathan G, ''Effectiveness and Efficieny of RFID technology in supply chain management: strategic values and challenges", Journal of Theoretical and Applied Electronic in Commerce Research, vol. 3, no. 2, pp 71-81, 2008.

[3] Applebaum, B., Freedman, M., Ringberg, H., Caesar, M., Rexford J.: Collaborative, Privacy-Preserving Data Aggregation At Scale (2009), http://eprint.iacr.org/2009/180.pdf

[4] Assaf Ben-David , Noam Nisan , Benny Pinkas, FairplayMP: a system for secure multi-party computation, Proceedings of the 15th ACM conference on Computer and communications security, October 27-31, 2008, Alexandria, Virginia, USA [doi>10.1145/1455770.1455804]

[5] Michael Ben-Or , Shafi Goldwasser , Avi Wigderson, Completeness theorems for non-cryptographic fault-tolerant distributed computation, Proceedings of the twentieth annual ACM symposium on Theory of computing, p.1-10, May 02-04, 1988, Chicago, Illinois, United States [doi>10.1145/62212.62213]

[6] Burkhart, M., Strasser, M., Many, D., Dimitropoulos, X.: SEPIA: Security through Private Information Aggregation (2009), http://arxiv1.library.cornell.edu/abs/0903.4258

[7] Wenliang Du , Mikhail J. Atallah, Secure multi-party computation problems and their applications: a review and open problems, Proceedings of the 2001 workshop on New security paradigms, September 10-13, 2001, Cloudcroft, New Mexico [doi>10.1145/508171.508174]

[8] European Commission. Report on Community Customs Activities on Counterfeit and Piracy (2008).

[9] Philippe Flajolet , Danièle Gardy , Loÿs Thimonier, Birthday paradox, coupon collectors, caching algorithms and self-organizing search, Discrete Applied Mathematics, v.39 n.3, p.207-229, Nov. 11, 1992 [doi>10.1016/0166-218X(92)90177-C]

[10] Goldreich, O.: Secure Multi-party Computation (2002), www.wisdom.weizmann.ac.il/~oded/pp.html

[11] O. Goldreich , S. Micali , A. Wigderson, How to play ANY mental game, Proceedings of the nineteenth annual ACM symposium on Theory of computing, p.218-229, January 1987, New York, New York, United States [doi>10.1145/28395.28420]

[12] Carmit Hazay , Yehuda Lindell, Efficient protocols for set intersection and pattern matching with security against malicious and covert adversaries, Proceedings of the 5th conference on Theory of cryptography, March 19-21, 2008, New York, USA

[13] Kerschbaum, F.: Practical Privacy-Preserving Benchmarking. In: Proceedings of the 23rd IFIP International Information Security Conference (2008).

[14] Ziekow Holger, Ivantysynova Lenka, Gunther Oliver, Arhitectural Patterns for RFID Applications in Manufacturing, 18th European Conference on Information Systems, Manuscript ID: ECS2010-0325, 2010.