Modelling Secure IT Systems–A Survey

Johannes Grünbauer, Jan Jürjens, Guido Wimmel
Software & Systems Engineering, Institut für Informatik, TU München
{gruenbau|juerjens|wimmel}@in.tum.de

Motivation

The development of security-critical systems is difficult. Many systems are designed and realised which exhibit severe shortcomings, which sometimes enable spectacular exploits. In this short paper we give a brief overview on work at the Software & Systems Engineering group at TU München regarding this problem.

UMLsec

The Unified Modelling Language (UML, [RJB99], the de facto industrial standard in object-oriented modelling) offers an unprecedented opportunity for high-quality critical systems development that is feasible in an industrial context.

The advantages of UMLsec are, that a large number of developers is trained in UML. Furthermore, compared to previous notations with a user community of comparable size, UML is relatively precisely defined. In the end, there exist many tools or are being developed to construct and analyse UML models, to generate code, etc.

We use stereotypes, tags and constraints to encapsulate knowledge on prudent security engineering and thereby make it available to developers which may not be specialised in security. Some examples for the use of UMLsec diagrams are as follows:

Use case diagrams are commonly used to describe typical interactions between a user and a computer system in requirements elicitation. They can be used to capture security requirements.

Activity diagrams can be used to model workflow and to explain use cases in more detail. Similarly, they can be used to make security requirements more precise. The control flow within activity diagrams can depend on security requirements.

Deployment diagrams are used to describe the physical layer of a system. We use them to check whether the security requirements on the logical level of the system are enforced by the level of physical security, or whether additional security mechanisms (such as encryption) have to be employed.

Statechart diagrams, showing the changes in state throughout an object's life, can be used to specify security requirements on the resulting sequences of states and the interaction with the object's environment.

For more details, see [Jü04].

AUTOFOCUS

Software-Engineering of security-critical systems can be supported by the tool AUTOFOCUS. AUTOFOCUS is a CASE tool for graphically specifying distributed systems. It is based on the formal method FOCUS, and its models have a simple, formally defined semantics. AUTOFOCUS offers standard, easy-to-use description techniques for an end-user who does not necessarily need to be a formal methods expert, as well as state-of-the-art techniques for validation and verification of the modelled systems. Systems are specified in AUTOFOCUS using static and dynamic views, which are conceptually similar to those offered in UML-RT. To be able to model, verify and validate security-critical systems, security aspects were included. AUTOFOCUS models should be used in security-critical systems development if a high degree of formality and tool support is required. Designers are thus forced to use a restricted set of description techniques and learn the (however intuitive) AUTOFOCUS notation. See [AF] for more information and a tutorial.

Secure Systems Development Based on the Common Criteria

The Common Criteria for IT Security Evaluation (CC) is an international standard for the assessment and certification of the security of an IT system. To obtain an evaluation according to the CC, one has to fulfil a number of requirements (such as models of different degree of formality, testing, configuration and life cycle management etc.) in the development of the system to be certified. The CC does not specify at which point in development the activities to fulfil these requirements must be carried out.

Case Studies

To demonstrate these techniques, there exist several case studies. Several case studies for UMLsec can be found in [Jü04]. [Ve01, VWW02] describes a process model for secure systems development based on the CC, which suggests appropriate points in development to fulfil the requirements of the CC, in context of a phase oriented process. The process has been applied in the case study PalME. A special application of the modelling tool AUTOFOCUS in the field "Security Engineering" can be found in [GHJW03]. Here, two security protocols have been modelled in AUTOFOCUS and model checked for correctness.

References

[AF] AUTOFOCUS homepage. URL: `http://autofocus.in.tum.de`.

[GHJW03] Grünbauer, J., Hollmann, H., Jürjens, J., und Wimmel, G.: Modelling and verification of layered security-protocols: A bank application. In: *SAFECOMP'03*. LNCS. Springer. 2003.

[Jü04] Jürjens, J.: *Secure Systems Development with UML*. Springer. March 2004. To be published.

[RJB99] Rumbaugh, J., Jacobson, I., und Booch, G.: *The Unified Modeling Language Reference Manual*. Addison-Wesley. 1999.

[Ve01] Vetterling, M.: Security Engineering nach den Common Criteria — eine Fallstudie. Master's thesis. Technische Universität München. August 2001.

[VWW02] Vetterling, M., Wimmel, G., und Wißpeintner, A.: Secure Systems Development Based on the Common Criteria. In: *10th International Symposium on the Foundations of Software Engineering (FSE-10)*. 2002.