

## **A methodology to build secure systems using patterns**

Eduardo B. Fernandez, Maria M Larrondo-Petrie, and Michael VanHilst  
Dept. of Computer Science and Eng., Florida Atlantic University, Boca Raton, FL 33431

Most of the approaches to produce secure software are based on analyzing code. While this is a reasonable approach, it will not have a strong impact in future systems. We believe that we need to emphasize the modeling aspects of code development and we have proposed a methodology for this purpose. We described this methodology at last year's ACSAC. This is an update of the work performed this year. A main idea in the proposed methodology is that security principles should be applied at every stage of the software lifecycle and that each stage can be tested for compliance with security principles [Fer06a]. Another basic idea is the use of patterns to guide security at each stage [Sch06]. Patterns are applied in the different architectural levels of the system to realize security mechanisms. This project proposes guidelines for incorporating security from the requirements stage through analysis, design, implementation, testing, and deployment. We discuss each stage indicating the most recent work. Modeling can include also hardware, which means that a complete secure system can be designed in this way.

*Domain analysis stage:* A generic conceptual model is defined. Legacy systems are identified and their security implications analyzed. Domain and regulatory constraints are identified. Analysis patterns lead to a domain model. Institution security policies are defined now but specific application policies are added later. The suitability of the development team is assessed, possibly leading to added training. Security issues of the developers, themselves, and their environment may also be considered in some cases. This phase may be performed only once for each new domain or team.

*Requirements stage:* Use cases define the required interactions with the system. Applying the principle that security must start from the highest levels, it makes sense to relate attacks to use cases. We study each activity within a use case and see which threats are possible [Fer06b]. We then determine which policies would stop these attacks. From the use cases we can also determine the needed rights for each actor and thus apply a need-to-know policy. Note that the set of all use cases defines all the uses of the system and from all the use cases we can determine all the rights for each actor. The security test cases for the complete system are also defined at this stage.

*Analysis stage:* Analysis patterns can be used to build the conceptual model in a more reliable and efficient way. We build a conceptual model where repeated applications of a security model pattern [Fer07] realize the rights determined from use cases. In fact, analysis patterns can be built with predefined authorizations according to the roles in their use cases [Fer07]. Then we only need to additionally specify the rights for those parts not covered by patterns.

*Design stage:* Design mechanisms are selected to stop the attacks identified earlier and realize the required policies [Fer05]. User interfaces should correspond to use cases and may be used to enforce the authorizations defined in the analysis stage. Secure interfaces enforce authorizations when users interact with the system. Components can be secured by using authorization rules for Java or .NET components. Distribution provides another dimension where security restrictions can be applied. Deployment diagrams can define secure configurations to be used by security administrators. A multilayer architecture is needed to enforce the security constraints defined at the application level. In each level we use patterns to represent appropriate security mechanisms. Security constraints must be mapped between levels.

*Implementation stage:* This stage requires reflecting in the code the security rules defined in the design stage. Because these rules are expressed as classes, associations, and constraints, they can be implemented as classes in object-oriented languages. In this stage we can also select specific security packages or COTS, e.g., a firewall product, a cryptographic package. Some of the patterns identified earlier in the cycle can be replaced by COTS (these can be tested to see if they include a similar pattern).

### **Current and future work**

We developed a new type of pattern, the attack pattern, that describes the steps of a known exploit in a given environment, e.g. DoS in VoIP [Fer07e]. We showed how to superpose security patterns on analysis patterns to produce secure analysis patterns, that incorporate security policies to stop specific types of attacks [Fer07d]. We developed more security patterns, in particular patterns for distributed systems [Del07], and VoIP [Fer07f]. Two years ago we started to describe standards as patterns. In this way we can make complex standards easier and we can compare standards. We can also see if a product specification complies with the standard. We developed patterns for H.323, SIP [Pel07], and WiMax [Fer07g]. We are

incorporating our work in an MDA (Model Driven Architecture) framework [Bro04]. In particular, we are looking at model mappings between architectural levels through the use of patterns.

## References

- [Del07] N. Delessy, E.B.Fernandez, M.M. Larrondo-Petrie, and J. Wu, "Patterns for access control in distributed systems", accepted for the *14th Pattern Languages of Programs Conference (PLoP2007)*, Monticello, Illinois, USA, September 5-8, 2007.
- [Fer06a] E.B. Fernandez, M.M. Larrondo-Petrie, T. Sorgente, and M. VanHilst, "A methodology to develop secure systems using patterns", Ch5 in *"Integrating security and software engineering: Advances and future vision"*, H. Mouratidis and P. Giorgini (Eds.), IDEA Press, 2006, 107-126.
- [Fer06b] E.B.Fernandez, M. VanHilst, M.M.Larrondo-Petrie, and S. Huang, "Defining security requirements through misuse actions", *Procs. of the International Workshop on Advanced Software Engineering (IWASE 2006)*. Santiago, Chile, August 2006 (part of WCC)
- [Fer07a] E.B.Fernandez, P. Cholmondeley, and O. Zimmermann, "Extending a secure system development methodology to SOA", *Procs. of the 1st Int. Workshop on Secure Systems Methodologies Using Patterns (SPattern'07)*.
- [Fer07b] E.B.Fernandez, J. Ballesteros, A. C. Desouza-Doucet, and M.M. Larrondo-Petrie, "Security Patterns for Physical Access Control Systems", in S. Barker and G.J. Ahn (Eds.), *Data and Applications Security XXI*, LNCS 4602, 259-274, Springer 2007. *Procs. of the 21st Annual IFIP WG 11.3 Working Conference on Data and Applications Security*, Redondo Beach, California, U.S.A, July 8-11, 2007.
- [Fer07c] E. B. Fernandez, D. L. laRed M., J. Forneron, V. E. Uribe, and G. Rodriguez G. A secure analysis pattern for handling legal cases", *Procs. of the 6th Latin American Conference on Pattern Languages of Programming ( SugarLoafPLoP'2007)*.
- [Fer07d] E.B.Fernandez and X.Y. Yuan, " Securing analysis patterns", *Procs. of the 45th ACM Southeast Conference (ACMSE 2007)*, March 23-24, 2007, Winston-Salem, N. C. 2007.
- [Fer07e] E. B. Fernandez and M. M. Larrondo Petrie, "Securing design patterns for distributed systems", Chapter 3 in *"Security in Distributed, Grid, and Pervasive Computing"*, Y. Xiao (Ed.). Auerbach, CRC Press, 2006, 53-66.
- [Fer07e] E.B. Fernandez, J.C. Pelaez, and M.M. Larrondo-Petrie, "Attack patterns: A new forensic and design tool", *Procs. of the Third Annual IFIP WG 11.9 Int. Conf. on Digital Forensics*, Orlando, FL, Jan. 29-31, 2007. [www.cis.utulsa.edu/ifip119](http://www.cis.utulsa.edu/ifip119)
- [Fer07f] E.B.Fernandez, J.C. Pelaez, and M.M. Larrondo-Petrie, "Security patterns for voice over IP networks", *Procs. of the 2nd IEEE Int. Multiconference on Computing in the Global Information Technology (ICCGI 2007)*, March 4-9, Guadeloupe, French Caribbean.
- [Fer07g] E. B. Fernandez , M. VanHilst, and J.C. Pelaez, "Patterns for WiMax security", *Procs. EuroPLoP 2007*, <http://hillside.net/europlop/home.html>
- [Pel07] J. C. Pelaez, E.B.Fernandez, and C. Wieser, "Patterns for VoIP signaling protocol architectures", *Procs. EuroPLoP 2007*. <http://hillside.net/europlop/home.html>
- [Sch06] M. Schumacher, E.B.Fernandez, D. Hybertson, F. Buschmann, and P. Sommerlad, *Security Patterns: Integrating security and systems engineering*, Wiley 2006.