1. Introduction

Electronic commerce and similar applications in the Internet have security as a fundamental requirement. The recently increasing occurrence of severe attacks has shown, however, that we still need quite some time and effort to reach sufficient security levels in IT systems. At present, there is a large gap between theory and practice in this area. Better ways to develop secure software, based on appropriate use of theory, are necessary. A promising approach is the use of patterns. Patterns have proved their value for the development of software. One of the current problems is that software developers do not know much about security and patterns could be an easy way to improve their understanding of security issues.

A Security Pattern describes a particular recurring security problem that arises in specific contexts and presents a well-proven generic scheme for its solution. In particular security patterns have several advantages:

- Novices can act as security experts.
- Security experts can identify, name and discuss both problems and solutions more efficiently.
- Problems are solved in a structured way.
- Dependencies of components can be identified and considered appropriately.

Several contributions on Security Patterns have been published within the last few years. However, there are some open issues that are still left for future work. The main objective of this workshop has been to bring together security patterns enthusiasts, to establish a forum for security patterns and to improve the overall work on security patterns.

The following people submitted position and pattern papers (without a particular order): Juha Pärsinnen, Sami Lehtonen, Ben Elsinga, Duane Hybertson, Markus Schumacher, Eduardo Fernandez, Manuel Görtz, and Aaldert Hofman. Besides, the following people visited and contributed to the workshop: Uwe Zdun, Oliver Vogel, Gaynor Redvers-Mutton, and Frank Buschmann.

2. Toward an integrated Security Pattern System

A primary goal of our efforts is to provide a unified perspective on the state of the art of security patterns and their application. In the following we present results of the first meeting of the slowly but steadily growing security pattern community.

2.1. Perception on Security Patterns

Aaldert Hofman contributed the following nice story about the perception on security patterns:

The perception of what a security pattern might be or look like can vary enormously between people. Working in a project on security patterns, one of the authors interviewed three people from the organization separately. All of them had more or less the same experience in their own organization and were working in projects for years.

When the author asked them about the number of security patterns they foresaw in their own organization their answers were remarkable. "Oh, a couple of hundred, I think" said the program manager. "Well, I reckon about 30 or 40" was the answer of the lead architect. "How many patterns do I see? Four, perhaps five, at most" replied the business manager.
Remember these people all knew their organization very well and had the same experience. But they had a different perception of patterns. During the interviews it became clear that the business manager was thinking of completely reusable parts of business processes, applicable without any further adjustment to specific circumstances. The lead architect thought of specific solutions like the use of smart cards, to be reused within several parts of the organization. Finally, the program manager was the opinion that every reusable part of program coding was a pattern.

Morale? Do not assume that all people share the same definition. Furthermore, don't assume that people share your definition.

2.2. Security Pattern Template

We decided to follow the template introduced in the POSA book. The key problems solved with Security Patterns are threats (or specific attacks leading to them) that occur in specific contexts. In the solution, one or more countermeasures mitigate the damage of such threats. In particular the template will be as follows:

<table>
<thead>
<tr>
<th>Name</th>
<th>A NounPhraseName related to the solution, i.e. what is being built. In addition a short summary could be provided.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alias</td>
<td>Other names for the pattern, if any are known.</td>
</tr>
<tr>
<td>Example</td>
<td>A running example/scenario/war story that demonstrates the existing problem and motivates the pattern. Possible attacks fit in here very well.</td>
</tr>
<tr>
<td>Context</td>
<td>The situation in which the pattern can be applied. This contains references to patterns that were applied before and led to this context.</td>
</tr>
<tr>
<td>Problem</td>
<td>A short and precise problem statement. This includes threats and attacks that lead to the problem.</td>
</tr>
<tr>
<td>Forces</td>
<td>Forces that are associated with the context/problem</td>
</tr>
<tr>
<td>Solution</td>
<td>The rule/design/principle that solves the problem and balances the forces.</td>
</tr>
<tr>
<td>Implementation</td>
<td>Guidelines for implementing the pattern</td>
</tr>
<tr>
<td>Example Resolved</td>
<td>How has the world changed after applying the pattern</td>
</tr>
<tr>
<td>Consequences</td>
<td>Benefits and liabilities of the pattern (related to the forces, new problems, etc.)</td>
</tr>
<tr>
<td>Known Uses</td>
<td>Examples of the use of the pattern taken from existing systems.</td>
</tr>
<tr>
<td>See Also</td>
<td>References to related patterns (variants, more specialized versions, required patterns, refinement patterns, etc.) The type of relationship has to explicitly explained</td>
</tr>
</tbody>
</table>

2.3. Organization Scheme

We found out that there are different views on security:

- Duane Hybertson suggested two organization schemes in the position paper of his group. One is an extension of the Zachmann framework by an architectural security view that focuses on scope (planner), business model (owner), system model (designer), technology model (builder) and detailed representations (subcontractor). At least the business, system and technology models are in line with the organization scheme proposed by Aaldert Hofman. He figured this out by asking different people how many security patterns exist in their environment.
• Duane and his colleagues also stated that views exist within and across the three dimensions composition, generalization and representation.

• Other suggestions were to use the categories applied in ISO 17799 and to use the lifecycle as an attribute for classification.

A requirement for the book will be to use a comprehensive scheme. Too many dimensions may be too confusing for such a document. However, we felt free to make use of all of these approaches to find out where “clusters” of security patterns emerge. Then we stripped down the organization scheme to a sound subset of possible attributes. This can be found in Section 3.

2.4. Forces
We identified different types of forces that typically appear in the security domain:

• Threats (and particular attacks)
• Policy Enforcement
• Security vs *ilities such as functionality, reliability, usability, efficiency, maintainability, and portability (refer to ISO 9126 – Software product quality).
• Increasing dependence on IT
• Security (un)awareness
• Time-based security concepts (suggested by Ben Elsinga)

2.5. Expectations on an integrated Security Pattern System
In a brainstorming session we asked each participant, what he expects from the book, especially how are these security patterns useful for the reader, i.e. what is the added value? A summary of the answers follows in chronological order:

• Making the relationships to other security approaches clear (e.g. attack trees, security standards, semantic analysis patterns, misuse cases, etc.)
• Show the integration of security patterns with the design by example (a generic and complete approach might remain a lifetime desire).
• Educational aspects, teaching security knowledge and skills to novices
• Promotional aspects (Ben Elsinga’s vision to think of Bill Gates saying “We used these security patterns and suddenly saw the light.”)
• Access knowledge in a convenient way (compared to textbooks)
• Complement knowledge over time (Pattern approach), accompanying web-site
• The book serves as an overview (experts) and introduction (novices). Running examples clarify how to apply the security patterns.
• Provision of case studies

3. Roadmap
We also worked on an overall landscape of security patterns. In particular, we identified three categories of security patterns: enterprise patterns, architecture/design patterns, and usage patterns. It can happen that patterns of a given pattern language are scattered across several categories. It is left for future work to show sequences of using these patterns. Otherwise the “generative” aspect of the original work might be lost.

• Security patterns at the enterprise layer are mainly about enterprise level issues such as security management, security principles, institutional security policies (e.g. need-to-know), and enterprise needs for various security properties and services – confidentiality, integrity, availability, accountability, I&A, access control, audit.

• We also consider architectural/design security patterns, which in general are system level solutions that respond to the enterprise level security policies and needs. This is the most important level for facilitating building security into a system. Examples include when to use appropriate services,
what the impacts will be, how to manage cryptographic keys, how to achieve network protection with firewalls, how to apply security in operating systems, etc. We also show system security principles and policies, such as design principles, models for access control, and execution policies.

- Security is often a matter of appropriate deployment, administration, and operation. Often products are shipped with disabled security capabilities. Therefore it is important to show how security solutions are used and operated properly. Also, how to define appropriate access rights for users and roles. These patterns show how to achieve security in an operational context, across all security categories.

The purpose of the case studies is to illustrate how to use the patterns to incorporate security into enterprise and systems engineering. We plan to show the use of patterns in specific domains, including

- Wireless LAN
- IP Telephony
- Sessions
- RADIUS authentication
- Distributed systems

Both Wireless LAN and IP Telephony are rather new areas. Although they adopt/build on well-known techniques/concepts we have observed that typical errors are made again (e.g., default passwords, etc.). Security patterns can help avoid such errors at different layers. Sessions can be found in many protocols. With the Session Pattern Language and the Distributed systems framework we can show how standard software patterns can be combined with security patterns.