Abstract: Security needs to be implemented in many applications in order to prevent unauthorized users from accessing them. However, usability is an issue that many current security solutions fail to address. This paper discusses major problems concerning usability in existing security tools and identifies new research directions to solve them.

Many security tools have been developed, but their successful use in real applications is quite limited. Among the primary reasons are their complexity in usage, requirement of prior advanced technical knowledge on the part of users and lack of useful visualization functionality. The first major problem - complexity in usage - discourages not only novice users but also application programmers and system administrators. This stems from the fact that too many steps and parameters need to be set in existing security tools in order for security services to be executed. Such complexity is often seen as obstacles of useful work or causes of poor system performance. This problem leads to the undesirable situations in that some users are willing to sacrifice security to meet their project deadlines or to achieve higher system performance. For those users who are willing to employ available security tools, the complexity can lead to unexpected consequences, as they may not understand the tools correctly. Research and development of techniques to reduce complexity in usage of security services are thus needed.

The second major problem with existing security tools is that they require users to be knowledgeable in security concepts, how security services work and their consequences. Unfortunately, the current IT workforce is not adequately trained in security. The evidence of this is the current computer science and engineering curriculum in many academic institutions where one often finds that not only security courses are not required, but also no security concepts are introduced or practiced in software engineering teaching. The gap between the knowledge required by the current security tools and the non-existing or limited security knowledge of the current IT workforce makes it difficult, if not impossible, for security services to be carried out successfully in many real-life applications. To solve this problem, besides addressing the education and training issues, research and development are needed of security tools that do not require users to have substantial technical knowledge in order to use them and are adaptive to users’ levels of expertise.

The third major problem is the lack of useful visualization functionality in today’s security tools. While a number of security visualization software packages have been
developed, many of them do not have the capability of displaying all system components at various states and various security checkpoints with an effective user-interface. It is important to research and develop security visualization tools which allow users to view what is going on in the system not only at the checkpoints that users set by themselves but also at the checkpoints that the system dynamically deems critical. Users should also be provided with a way to understand and deal with large amounts of tracking data from multiple sources, such as network management systems, application services, and sensor and mobile nodes, in a variety of formats and semantics that may even be conflicting with each other. Some of these heterogeneous data, especially sensor data, may also be corrupted or missing; so how to model, visualize and interpret data uncertainty should be provided. Users should also be presented with different scenarios of attacks, corresponding complete or partial solutions and their consequences in the rest of the system. This implies that if a security patch is recommended, then users should be informed of the effects of the patch on individual system components. In addition, the visualization tools should be adaptive to users’ levels of expertise and cognition.

To lessen the severity of the three above-mentioned usability problems in existing security tools, another direction that should be considered is to research and develop an autonomic secure system where human is taken out of the loop as much as possible. Such a system would have the capabilities of self-configuring, self-optimizing, self-protecting and self-healing. To realize this, it is important to carry out research for ways to automatically and dynamically model, capture, store, retrieve, update and analyze data and patterns of, for example, system usage, attacks, system performance, failure and recovery. In addition, appropriate metrics need to be developed to measure not only the correctness and performance, but also the usability of security tools.