Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Systems Security Problems and Cultural Meanings in Control and Automation Systems: Empirical Evidence for Value Conflicts in Systems Engineering

Amanda Freeman*, Dr Larry Stapleton* & Dr Gabriel Byrne**

**ISOL Research Group, Waterford Institute of Technology*
***Smurfit Business School, University College, Dublin*

Abstract: Automation and control systems such as integrated enterprise information technologies and distributed telemedical system architectures require highly secure information processing environments to ensure that costly (and even fatal) errors do not occur. However, research into systems development methodologies shows significant gaps in the treatment of systems security. Many methodologies do not specifically include security and privacy considerations within their frame of reference. As a consequence, recent studies of information control and management systems have shown that, in a global context, many organisations are at a significant security risk. In this paper we examine five of the most common system engineering methodologies cited in the literature, and we examine to what extent each of these methodologies incorporates security and privacy as part of the systems development process. This paper also presents empirical evidence to support the proposition that, as regards system security, high technology engineering education is at odds with the core values of students of systems engineering. This has major implications for the development of secure systems in a globalised economic context. The evidence shows how students from a wide variety of cultural backgrounds come into degree programmes valuing security highly, but the education programmes do not address systems security in very much depth.

1. THE IMPORTANCE OF HUMAN VALUES

There has been a growing interest in recent years in the analysis of human values (Kluckhon, 1951; Rokeach, 1973; Schwartz and Bilsky, 1987, 1990). Schwartz (1990, 878) defines values as "concepts or beliefs, that pertain to desirable end states or behaviours, they transcend specific situations, guide selection or evaluation of behaviour and events, and are ordered by relative importance". Values play a key role in human activity as they provide strong explanations of behaviour. Knowing persons values enables us to predict how he or she will behave in various experimental and real-life situations (Rokeach, 1973). Values also play a key role in our decision making. Research into values has provided valuable insights into individual, group and organisational levels of analysis. Schlienger and Teufel (2003) believe that every organisational culture expresses core values that are shared by the majority of the organisations members. Sarros and Santora (2001) assert that organisations with strong cultures and clear values increase their chance of success and longevity. Barrett (1998) attributes a strong organisational culture to be one where values are shared amongst staff and management.

Whilst many researchers have mentioned the role of values in systems engineering development (Hedberg and Mumford, 1975; Kling, 1978; Dagwell and Weber, 1983; Kumar and Welke, 1984; Mumford, 2000; McGuire et al., 2006) there is a paucity of academic research that examines values in terms of systems engineering security development.

The objective of this paper is demonstrate, how system engineering methodologies are out-of-step with the security and privacy values of young technologists from different cultures. In order to achieve this objective the following research question will be addressed: Do current systems engineering methodologies reflect the values of young technologists in terms of security and privacy values?

2. SYSTEMS ENGINEERING SECURITY DEVELOPMENT

The protection of information plays a vital role in automation and control technologies. However, research shows that companies are still suffering from serious security breaches (PricewaterhouseCoopers, 2005; CSI/FBI, 2005, Freeman and Doyle, 2006). Many researchers argue that a significant amount of these breaches are caused by poor software design (McGraw and Wyk, 2005; Villarroel et al., 2004). One of the reasons given for this is that security is generally only considered in the systems engineering process after the system has been developed (Mouratidis et al., 2004; Villarroel et al., 2004). McGraw and Wyk (2005) argue that one of the reasons for this is that system engineers are not security specialists. Dhillon (1995) and Ghosh et al., (2002) believe that in order for a system to be secure it is vital that system engineers work with security specialists. According to Baskerville (1993, 377) "what is missing in many instances is the involvement of concerned users, enlightened developers, experienced security specialists or other parties who understand how to incorporate controls into systems while the systems are still in development".

Therefore it is vital that systems security engineering development involves all the stakeholders within the business.

One novel way of analysing the role of stakeholders in system security engineering is to look at the values of each group involved. By making value stances explicit it helps us to interpret the meanings various groups have assigned to particular issues relating to system security engineering (Kling 1978). For example it is widely accepted that both management and staff play a key role in the implementation and success of a system (Dhillon and Backhouse, 2000; Dhillon, 2001; Freeman and Doyle, 2006).

Research shows that the type of methodology chosen by a system engineer tends to be based on their values (Kumar and Welke, 1984; Dhillon, 1995; Orvik et al., 1998). According to Baskerville (1993) advances in security methodologies lag behind advances in general system development methodologies. It is possible that a reason for this could be that security is not valued in the systems engineering community. In the following section we will give a brief account of some of the most widely reviewed system engineering methodologies in the literature and we will look at to what extent each of these methodologies incorporate security as part of the development process.

3. OVERVIEW OF SYSTEMS ENGINEERING METHODOLOGIES AND THE EXTENT TO WHICH THEY ADDRESS SECURITY ISSUES

3.1 SSADM

SSADM (Structured Systems Analysis & Design Methodology) is one of the most widely used 'hard' structured methodologies, and it is for this reason that it was chosen for review in this paper. It is said to be a data driven methodology as it places a large emphasis on data modeling and the database. It is a highly structured methodology that provides very detailed rules and guidelines to project developers. Version 4 of SSADM consists of five stages and each stage is made up of a series of steps which use appropriate techniques for the tasks involved. Each of the stages and steps has defined inputs and outputs (Avison and Fitzgerald, 1995). Goodland and Slater (1997) define the steps as; feasibility study, requirements analysis, requirements specification, logical systems specification and physical design. After each stage users are involved in formal quality assurance reviews, and informal walkthroughs where each stage is signed off before developers move on to the next stage. A weakness in SSADM as a methodology is that it does not provide any support for the planning, construction and implementation stages of development. The only emphasis placed on security in the SSADM methodology is in the requirements specification stage. However, security is only mentioned briefly in terms of access privileges and unauthorised access (Goodland and Slater, 1997).

3.2 Object Orientated (OO) Methodology – (UML/UP)

The essence of object-orientated analysis and design is to emphasise the problem domain and logical solution from the perspective of objects (things, concepts, or entities) (Larman, 1998, 6).

According to Dennis et al., (2006) any object-orientated approach to developing information systems must be, use case driven, architecture centric, and iterative and incremental. The OO methodology focused on in this paper is the Unified Modelling Language (UML) Unified Process (UP). Carew and Stapleton (2004, 85), describes UML as "a graphical modelling language for specifying systems from an object-orientated perspective". UML itself is not a methodology; it is a modelling notation that provides a variety of modellling diagrams, but it does not stipulate underlying process for developers to follow. Nonetheless, the authors of UML have provided the Unified Process (UP) as a suitable methodology. Bruegge and Dutoit (2000) describe the five types of notation used in UML as, use case diagrams which are used at the requirements elicitation stage to represent the functionality of the system, class diagrams which are used to describe the structure of the systems, sequence diagrams which are used to formalise the behaviour of the system, state diagrams are used to describe the behaviour of an individual object as a number of states and transition between these states and finally, activity diagrams which describe the system in terms of activities. Unlike the other methodologies mentioned in this paper UML does take security into account in the systems design phase and also in the testing phase. However, it only focuses on security in terms of authentication, access controls and encryption.

3.3 Soft Systems Methodology

The Soft System Methodology was developed by Peter Checkland as a way to deal with problem situations in development where there is a high social, political and human component involved. It is a seven stage/4 activities systems thinking approach for unstructured problems in the real world (Checkland, 1999). What makes the Soft System Methodology different from most other methodologies is that it focuses on 'soft' problems in systems development as opposed to 'hard' problems that are more technically orientated. Checkland (1999) argues that it is easy to model data and processes, but to understand the real world it is essential to include people in the model. In 1990 Checkland revised the SSM 7 Stage Model and presented the four-activities model of SSM. Essentially both models are similar in that they both focus on the 'soft' aspects of systems development. This methodology focuses on the 'soft' side of systems development and there is no specific reference to security.

3.4 Multiview

Multiview is described by Iivari (2000, 199) as a "methodology that explicitly attempts to reconcile ideas from several information systems development approaches, most notable the Soft System Methodology (SSM)". As a methodology it looks at both the human and technical aspects of ISD. There are five stages in the Multiview methodology and according to Avison and Wood-Harper (1990) these

stages aid in answering all the vital questions of users. The stages in Multiview are as follows:

Analysis of human activity, analysis of information, analysis and design of socio-technical aspects, design of the human computer interface and design of technical aspects. The first stage looks at the organisation itself, the second stage analyses the entities and functions of the system. Stage three includes user participation, to identify how the system can be fitted into the users' working environment. The fourth stage is concerned with the implementation of the computer interface. Avison and Wood-Harper (1990) believe this is a vital step as the way in which users interact with a system plays an important role in whether users' accept a system. Finally in the fifth stage the developer focuses on the efficient design and the production of a full system specification. Whilst security is mentioned in Multiview it is only briefly discussed and this discussion takes place as part of a case study.

3.5 ETHICS

Effective Technical and Human Implementation of Computer-based System (ETHICS) is a methodology developed by Enid Mumford. ETHICS takes the view that in order for technology to be successful it should fit closely with organisational and social factors (Avison and Fitzgerald, 1995). Mumford (2000) believes that an improved quality of working life and enhanced job satisfaction of the users must be a major objective of the systems design process. ETHICS is a seven stage methodology based on the participatory approach to information systems development. The seven stages are; diagnosis of user needs, setting efficiency and job satisfaction objectives, developing alternative design strategies, strategy selection to achieve objectives, hardware and software selection, implementation and systems evaluation (Mumford, 1990). Participation plays a role in many methodologies, but it plays a vital role in ETHICS. The role of the developer in ETHICS is very different to the role of the developer in the previously mentioned methodologies, in that the developer together with the user develops the systems. Security is not mentioned in the ETHICs methodology.

This section gave a broad overview of some of the most commonly cited system engineering methodologies that are covered in the literature. Looking at these methodologies we can see that with the exception of SSADM and the Object-Orientated methodology security is not considered.

Form this we can see that if systems engineers tend to value the less structured methodologies there is a greater chance that security will not feature in the development process. This may help to explain why security is often only considered in the system engineering process after the system has been developed. In order to understand how values can be explored the next section reviews the literature on measuring personal values.

4. MEASURING PERSONAL VALUES IN SYSTEMS ENGINEERING

Based on the work of Rokeach (1973), Schwartz and Bilsky (1987) devised a theory of universal types of values as criteria by viewing values as cognitive representations of three universal requirements which are, biologically based needs of the organism, social interaction requirements for interpersonal coordination and social institutional demands for group welfare and survival.

In order to measure values of individuals Schwartz developed the Schwartz's Value Survey (SVS) (Schwartz, 1992). ". In the SVS 57 values are used to represent 10 motivationally distinct value domains that are theoretically derived from universal requirements of human life, which are, Power (social power, authority, wealth), Achievement (success, capability, ambition, influence on people and events), Hedonism (gratification of desires, enjoyment in life, self-indulgence), Stimulation (daring, a varied and challenging life, an exciting life), Self-Direction (privacy, creativity, freedom, curiosity, independence, choosing one's own goals), Universalism (broad mindedness, beauty of nature and arts, social justice, a world at peace, equality, wisdom, unity with nature, environmental protection), Benevolence (helpfulness, honesty, forgiveness, loyalty, responsibility), Tradition (respect for tradition, humbleness, accepting one's portion in life, devotion, modesty), Conformity (obedience, honouring parents and elders, self-discipline, politeness) and Security (national security, family security, social order, cleanliness, reciprocation of favours (Lindeman and Versasalo, 2005). Results of these 10 domains have shown to load two bipolar dimensions; Conservation (whether people resist change and emphasise self-restriction and order) versus Openness to Change (whether people are ready for new experiences and emphasise independent action and thought) and Self Transcendence (whether people are willing to transcend selfish concerns and promote the welfare of others) versus Self Enhancement (whether people are more motivated to enhance their own personal interests even at the expense of others). These two dimensions reflect the different motivational goals of the 10 basic values and the two major conflicts that organise the whole value system (Lindeman,, 2005, 177).

One could argue that high achieving systems engineers would score high on Openness to Change (self-direction, stimulation etc.) versus Conservation (tradition, conformity, security (not to be confused with computer security)).

The personal values of Openness to Change include those values valued by all stakeholders of the development process namely: creativity, curious, choosing ones own goals, freedom, daring, varied life, exciting life. The personal value of privacy is also included in the value type 'Self-direction' suggesting that a high score on the bipolar value of Openness to Change and Self-direction would indicate systems engineers to be visionary types of people with a sense of self respect for the personal value of privacy.

This research we will be focusing on the personal value of privacy in terms of measuring security beliefs. When people

talk about security, they often mean data confidentiality. Clearly there is some relationship between security and privacy. In order to have privacy we must have security (Ghosh, 2001). If we want to protect information we have to ensure that the appropriate security measures are in place.

Lindeman and Verasalo (2005, 171) argue that, 'in many studies, a scale with 57 items may be too time consuming to fill in, and may take up too much space on a questionnaire'. With that in mind they developed a shorter version of the SVS, which they called the Short Schwartz's Value Survey (SSVS). In the SSVS, Lindeman and Verasalo (2005) set out to see if, by asking respondents to rate the importance of the ten values directly could Conservation and Self Transcendence be reliably and validly examined with a shortened version of the Schwartz's Value Survey.

Returning to the research question: Do current systems engineering methodologies reflect the values of young technologists in terms of security and privacy values? In order to answer this question a survey was conducted. The following section presents and interprets the findings of this study.

5. RESEARCH METHOD – PARTICIPANTS AND PROCEDURES

The sample consists of Irish computing students (78.9%) and (21.1%) of foreign students from the following countries, China, Pakistan, Nigeria, England, Italy, France, Sudan, Poland, Ukraine, Russia, Spain, Malawi, and South Africa . The sample consists of 161 participants representing a convenience sample, which is appropriate for a preliminary study of this nature. The respondents ranged from first year computing students up to computing master's students. The participants were told that the study concerned values and that participation was voluntary and that all information would be treated confidentially. Using the Short Schwartz's Value Survey participants were presented with the name of a value along with its value items. Participants were asked to rate the importance of each value as a guiding principle in their life. The 10 values were rated on a 9-point scale ranging from -1 (opposed to my principles), 0 (not important), 3 (important) to 7 (of extreme importance).

6. FINDINGS

Table 1 compares the mean scores for each of the 10 values in terms of Irish and Non-Irish technologists.

From this table we can see that the top four values that Irish technologists value as guiding principles in their life are: Self-direction (5.21), Benevolence (4.87), Achievement (4.74) and Security (4.17). In terms of foreign technologists the top for values they value as guiding principles in their life are: Self-direction (5.39), Security (4.82), Benevolence (4.79) and Universalism (4.50). These findings show us that participants do in fact value privacy, with Self-direction scoring highest with both Irish (5.21) and Non-Irish (5.39) participants.

Table 1: Mean Value Scores for Irish and Non-Irish Technologists

	Irish			Non-Irish		
Values	**N**	**Mean**	**Std. Dev**	**N**	**Mean**	**Std. Dev**
Power	124	2.87	2.453	32	2.81	2.278
Achievement	125	4.74	1.660	33	4.33	2.160
Hedonism	123	3.44	2.423	32	3.22	2.181
Stimulation	124	4.09	1.913	32	3.94	2.094
Self-Direction	126	5.21	1.641	33	5.39	1.519
Universalism	126	3.98	2.261	34	4.50	2.352
Benevolence	126	4.87	2.009	34	4.79	2.086
Tradition	125	2.43	2.315	33	3.06	2.738
Conformity	125	2.84	2.305	34	3.29	2.368
Security	125	4.17	2.003	33	4.82	1.722

-1 = opposed to my principles, 0 = not important, 3 important, 7 of supreme importance

Table 2 illustrates that the Irish technologists score quite high on Openness to Change (32.97%) and Conservation (24.43%) as do their Non-Irish counterparts. In terms of Self-Enhancement the Non-Irish technologists score slightly lower (17.78%) than the Irish participants (19.69%). In terms of Self-Enhancement there is very little difference between the Irish participants (22.9%) and the Non-Irish participants (23.13%). The fact that both groups scored highly on Openness to Change and Conservation, shows that, the technologists in both groups are independent thinkers who embrace new challenges but they also have a certain amount of order in what they do. In terms of Self-Enhancement both groups scored quite low, with the Non-Irish technologists valuing it even less than the Irish technologists. This would indicate that while these technologists do to some extent want to enhance their own personal interests it is not a huge motivating factor in what they do.

Table 2: Comparison of Bipolar Domains

	Openness to Change	**Conservation**	**Self-Enhance**	**Self-Trans**
Irish	32.97%	24.43%	19.69%	22.9%
Non-Irish	31.26%	27.82%	17.78%	23.13%

7. SYNTHESIS OF FINDINGS

In section 4 we asked the research question: Do current systems engineering methodologies reflect the values of young technologists in terms of security and privacy values? A review of five of the most commonly cited methodologies in the literature shows us that with the exception of SSADM and UML/UP security does not appear in any of the other three methodologies. This indicates that if a system engineer tended to favour the less structured methodologies there is a greater chance that security would not appear in the system

development process upfront. Also as the five methodologies investigated in this paper are commonly taught in colleges around the world. This shows us that high technology engineering education is possibly out-of-step with the core values of students in terms of security and privacy. This we believe has major implications for the development of secure systems in a globalised economic context. It could help to explain why some many companies still suffer significant security problems with their information control and management systems.

8. LIMITATIONS OF THE STUDY

In this paper we only examined five methodologies that are most commonly referred to in the literature. We acknowledge that in practice quite often organisations use a combination of different methodologies and we also acknowledge that in other cases some organisations may use no methodology or may use a methodology not mentioned in this paper. It is also important to highlight that there are many standalone security methodologies been developed regularly.

9. CONCLUSION

Developing and designing secure automation and control systems is a vital component of the systems engineering development process, yet security and privacy are often not considered until after the system has been developed. This reflects the low value that systems engineering places on security and privacy. The aim of this paper was to see if current engineering methodologies reflected young technologist's values in terms of security and privacy. We examined some of the most cited system engineering methodologies in the literature which are also some of the most commonly taught systems engineering methodologies in higher education. The findings show that regardless of cultural background the young technologists who took part in this study do in fact value security and privacy. We have shown that some of the most common system engineering methodologies do not even consider security as part of the development process. This we believe this shows that current high technology engineering education is out of step with the core values of students in systems engineering. And this we believe cause serious value conflicts for system engineers.

10. ACKNOWLEDGEMENTS

This research has been funded by the Irish Research Council for Science, Engineering and Technology (IRCSET).

11. REFERENCES

Avison, D. & Shah, H. (1997) *The Information Systems Development Life Cycle*, London, McGraw Hill.

Avison, D. E. & Fitzgerald, G. (1995) Information Systems Development: Methodologies, Techniques and Tools, New York, McGraw Hill.

Avison, D. E. & Wood-Harper, A. T. (1990) *Multiview: An Exploration in Information Systems Development*, London, McGraw Hill.

Barrett, R. (1998) *Liberating the Corporate Soul.* Butterworth-Heinemann, Woburn, MA.

Baskerville, R. (1993) Information Systems Security Design Methods: Implications for Information Systems Development. *ACM Computing Surveys*, **25**, (4), 376-414.

Bruegge, B. & Dutoit, A. (2000) Object-Orientated Software Engineering: Conquering Complex and Changing Systems, New Jersey, Prentice Hall.

Carew, P. & Stapleton, L. (2004) Towards a Privacy Framework for Information Systems Development. IN Caplinskas, A., Wojtkowski, G., Wojtkowski, W., Zupancic, J. & Wrycza, S. (Eds.) *Information Systems Development: Advances in Theory, Practice, and Education,* Lithuania, Springer.

Checkland, P. (1999) *Systems Thinking, Systems Practice,* New York, Wiley.

CSI/FBI (2005) Computer Crime and Security Survey 2005. Computer Security Institute.

Dagwell, R. & Weber, R. (1983) System Designers' User Models: A Comparative Study and Methodological Critique. *Communications of the ACM*, **26**, (11), 978-995.

Dennis, A., Wixom, B. & Roth, R. (2006) *Systems Analysis Design*. Wiley, New Jersey.

Dhillon, G. (1995) Interpreting the Management of Information System Security. *School of Economics and Political Science.* London, University of London.

Dhillon, G. (2001) Violations of Safeguards by Trusted Personnel and Understanding Related Information Security Concerns. *Computers and Security,* 20, (2), 165-172.

Dhillon, G. & Backhouse, J. (2000) Information System Security Management in the New Millennium. *Communications of the ACM,* 43, (7), 125-128.

Freeman, A. & Doyle, L (2006) An Exploratory Study Investigating the Role of Information Systems Security in SMEs in the South East of Ireland. *EUTIC06 Colloquium,* University of Brussels.

Ghosh, A., Howell, C. & Whittiaker, J. (2002) Building Software Securely from the Ground Up. *IEEE Software,* Feb 2002, 14-16.

Goodland, M. & Slater, C. (1997) *SSADM: A Practical Approach,* London, McGraw Hill.

Hedberg, B. & Mumford, E. (1975) The Design of Computer Systems: Man's Vision of Man as an Integral Part of the Systems Design Process. *Human Choices and Computers.* Amsterdam. North-Holland Publishing Company.

Iivari, J., Hirschheim, R. & Klein, H. K. (2001) A Dynamic Framework for Classifying Information Systems Development Methodologies and Approaches. *Journal of Information Systems,* **17***, (3), 179-219.*

Kling, R.. (1978) Value Conflicts and Social Choice in Electronic Funds Transfer System Developments. *Communications of the ACM,* **21**, (8), 642-657.

Kluckhohn, C., (1951) Values and Value-Orientation in the Theory of Action. In *Toward a General Theory of Action,* in Parsons, T., Shils, E. (Ed.), Harvard Business Press, Cambridege, MA.

Kumar, K. & Welke, R. J. (1984) Implementation Failure and System Developer Values: Assumptions Truisms and Empirical Evidence. The Fifth International Conference of Information Systems. Tuscon, Arizona.

Larman, C. (1998) Applying UML and Patterns: An Introduction to Object-Orientated Analysis and Design. New Jersey, Prentice Hall.

Lindeman, M. & Versasalo, M. (2005) Measuring Values With the Short Schwartz's Value Survey. Journal of Personality Assessment, 85, (2), 170-178.

McGraw, G. & Wyk, K. (2005) Bridging the Gap Between Software Development and Information Security. IEEE Security & Privacy. **3,** (5), 75.79.

McGuire, D., Garavan, T., Saha, S. & Donnell, D. O. (2006) The Impact of Individual Values on Human Resource Decision-Making by Line Managers. *International Journal of Manpower,* **27***, (3), 251-273.*

Moutatidis, H., Giorgini, P. & Manson, G. (2004) When Security Meets Software Engineering: A Case of Modelling Secure Information Systems, *Information Systems Journal,* 2005, (30), 609-629.

Mumford, E. (1990) *Designing Human Systems for New Technology*, Manchester Business School Press.

Mumford, E. (2000) A Socio-Technical Approach to Systems Design, *Requirements Engineering,* 125-133.

Orvik, T. U., Olsen, D. H. & Ssin, M. (1998) Deployment of System Development Methods. IN Zupancic, J., Wojtkowski, W., Wojtkowski, W. G. & Wrycza, S. (Eds.) *Evolution and Challenges in System Development.* Bled, Slovenia.

PricewaterhouseCoopers (2005) Information Security Breaches Survey 2005. PricewaterhouseCoopers.

Rokeach, M. (1973) *The Nature of Human Values*, London, Collier Macmillan Publishers.

Sarros, J. & Santroa, J. (2001) Leaders and Values: A Cross-Cultural Study. *Leadership and Organization Development Journal.*, **22,** (5), 243-248.

Schlienger, T. & Teufel, S. (2003) 'Analyzing Information Security Culture: Increased Trust by an Appropriate Information Security Culture'. *Proceedings of the International Workshop on Database and Expert Systems Applications* (DEXA'03).

Schwartz, S. H. (1992) Universals in the Content and Structure of Values: Theoretical Advances and Empirical Tests in 20 Countries. *Advances in Experimental Social Psychology,* **25**, 1-65.

Schwartz, S. H. & Bilsky, W. (1987) Toward a Universal Psychological Structure of Human Values. *Journal of Personality and Social Psychology*, **53**, (3), 550-562.

Schwartz, S. H. & Bilsky, W. (1990) Toward a Theory of the Universal Content and Structure of Values: Extensions and Cross-Cultural Replication. *Journal of Personality and Social Psychology*, **58,** (5), 878-891.

Villarroel, R., Medina, E. & Piattini, M. (2004) Incorporating Security Issues in the Information Systems Design. *Fifth Mexican International Conference in Computer Science (ENC'04).* IEEE Computer Security.