Safe and Sound: A Safety-Critical Approach to Security

Sacha Brostoff
Dept. of Computer Science, University College London
Gower Street
London, UK WC1E 6BT
+44 (0)20 7679 3039
s.brostoff@cs.ucl.ac.uk

M. Angela Sasse
Dept. of Computer Science, University College London
Gower Street
London, UK WC1E 6BT
+44 (0)20 7679 7212
a.sasse@cs.ucl.ac.uk

ABSTRACT

This paper firstly argues that the design of security applications needs to consider more than technical elements. Since almost all security systems involve human users as well as technology, security should be considered, and designed as, a socio-technical work system. Secondly, we argue that safety-critical systems design has similar goals and issues to security design, and should thus provide a good starting point. Thirdly, we identify Reason's (1990) *Generic Error Modeling System/Basic Elements of Production* as the most suitable starting point for a socio-technical approach, and demonstrate how its basic elements can be applied to the domain of information security. We demonstrate how the application of the model's concepts, especially the distinction between *active* and *latent* failures, offers an effective way of identifying and addressing security issues that involve human behavior. Finally, we identify strengths and weaknesses of this approach, and the requirement for further work to produce a security-specific socio-technical design framework.

1. INTRODUCTION

In recent years, the security research community has come to recognize the importance of the human factor in security. In an increasing number of cases, user behavior facilitated security breaches, prompting Schneier (2000) to state that:

> *"Security is only as good as it's weakest link, and people are the weakest link in the chain."*

The opposition recognized and exploited this state of affairs earlier. In his testimony to the US Senate committee hearing, Kevin Mitnick pointed out that he had obtained most passwords from unwitting users, rather than by cracking. In his lectures to IT managers, he has repeatedly emphasized that:

> *"The human side of computer security is easily exploited and constantly overlooked. Companies spend millions of dollars on firewalls, encryption and secure access devices, and it's money wasted, because none of these measures address the weakest link in the security chain."* (Poulsen, 2000).

The recognition that security involves people as well as technology is an important first step. However, labeling users as the "weakest link" implies that they are to blame for the current state of affairs. We argue that this is an unfortunate repeat of the "human error" perspective, which blighted the development of safety-critical systems in the mid-eighties: pilots and operators were blamed for accidents whenever they took a wrong action when dealing with a critical incident. Today, we know that pilots and operators are hardly ever careless. They are desperately trying to identify the right – life-saving – actions, but fail because (a) designs give wrong cues about the cause of the problem, (b) unattainable cognitive or physical tasks, (c) insufficient knowledge to identify the correct action, or (d) insufficient training to carry it out correctly. Adams & Sasse (1999) pointed out that security has largely failed to consider usability, and consequently, the demands security mechanisms make on users have been allowed to increase unchecked. In many environments, the demands that different security mechanisms place on users have become unattainable, or conflict with other elements of users' jobs. Furthermore, many users receive little or no training or support on security. These issues cause or facilitate security breaches; yet, they are not addressed by current security models.

2. WHY DO WE REQUIRE A NEW MODEL?

Traditionally, security models describe what the protection mechanism is to achieve (Anderson, 2001).

1. The **Bell-LaPadula** model (Bell & LaPadula, 1973) aims to enforce the principle of *confidentiality*, and describes the basic functions of a multilevel-secure system. There are two basic rules: (1) *no read up*, and (2) *no write down*. A process may not read data that is at a higher security classification than itself, and may not write data at a security classification lower than itself.
2. The **BIBA** policy model (Biba, 1977) aims to enforce the principle of *integrity*. It is an upside-down version of Bell-LaPadula, where processes may not read data of a lower classification than themselves, and may not write data to a higher level than themselves. Instead of being a constraint primarily on who can read something, it is a constraint on who can write or alter something.
3. The **Clark-Wilson** (Clark and Wilson, 1987) security model is a formalisation of banking and accounting

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
NSPW'01, September 10-13th, 2002, Cloudcroft, New Mexico, USA.
Copyright 2002 ACM 1-58113-457-6/01/0009...$5.00.

procedures, such as double-entry bookkeeping. It ensures, among other things, that transactions maintain balance, and that they can be reconstructed.

We argue that these models do not predict, or address security problems that can be observed in everyday use. Consider the following examples of problems with one specific security mechanism, passwords:

1. *Conflicting security and task demands.* Adams and Sasse (1999) found that users circumvent password mechanisms (sharing and disclosure of passwords) because they conflict with users' task demands and working practices. Models such as BLP rely on authentication being carried out correctly and legitimately, but they make no allowance for the need to adapt procedures to fit an individual's or organization's primary task.
2. *Unattainable cognitive demands.* With the increasing number of systems, users are increasingly unable to cope with the number of passwords or the rules that govern their use (Adams & Sasse, 1999); as a result, the cost of re-setting passwords has been escalating in many organizations. Existing security models such as BLP, Clark-Wilson and BIBA design security on a per-system basis, and do not address authentication for multiple systems.
3. *Lack of user training and support.* Many users in Adams and Sasse's (1999) study did not appreciate how crackers attempt to break into computer systems, and so chose easy to break passwords. Models such as BLP do not explain or predict this type of vulnerability, and so have little to contribute to its solution.
4. *Lack of security management.* 30% of managers in a large technology company reported that their last password reset was due to *"circumstances beyond their control"* (Sasse et al., 2001). Causes identified include (a) being away from base for long enough that their accounts were suspended, (b) server upgrades, and (c) automated remote disc mounting features of operating systems. Current security models do not cover management of authentication in longer-term, real-world use.
5. *Distribution of hardware and data.* Security problems resulting from lost or stolen laptops are common today, e.g. the laptop stolen after a presentation from the CEO of QUALCOMM (Lemos, 2000). Laptops belonging to key staff contain commercially sensitive data (e.g. relating to a potential merger or takeover). BLP does not apply here, because physical access to the hard disk can be used to bypass access control mechanisms. For BLP to work in this case, a mechanism would be needed that automatically encrypts any data transferred to a laptop or other external system (we hypothesize that many users would not use encryption voluntarily). However, most laptops are used with off-the-shelf operating systems that do not provide such security features.

These examples show how the current approach of designing security on a per-system basis creates a system where users – like pilots and operators in the eighties – are put in situations where they will inevitably fail. Sasse et al. (2001) conclude that designers of security systems need to recognize that security is a socio-technical system, and that all parts of the system and the way they interact need to be included to achieve effective security.

The concept of *survivability* (Lipson and Fisher, 1999) has a wider view than traditional security models, and could be applied to all of the scenarios presented above. However, survivability is concerned with what to do after the breach has occurred, not preventing it or its recurrence. A new model is still required, which will be complimentary to existing perspectives.

3. THE CASE FOR ADAPTING SAFETY MODELS FOR SECURITY

3.1 Similarities Between Safety and Security Domains

Both safety and security contain all of the *basic elements of production*, which are core components of the new model (see next section for a description of these components). Both safety and security are *secondary goals*: they exist to protect an organization and its staff while they are engaged in the primary task - *production*.

3.1.1 Economics

The goals of safety and security are compatible with the goals of production - *in the long-term*. To continue producing, an organization needs to be both safe and secure. Given that resources are finite, there are likely to be many occasions in which there are short-term conflicts of interest between production and either safety or security. Resources allocated toward production are likely to diminish those available for either safety or security, and vice versa. This resource allocation dilemma is exacerbated by:

1. **The certainty of outcome.** Resources aimed at improving productivity have relatively certain outcomes; those aimed at enhancing safety or security do not - at least in the short-term. This is because both safety and security breaches are due in large part to stochastic elements.
2. **The nature of feedback.** The feedback generated by the pursuit of production goals is generally unambiguous, rapid, compelling and highly reinforcing (when the news is good). The feedback associated with the pursuit of safety and security goals is largely negative, intermittent, often deceptive, and perhaps only compelling after a major accident or string of incidents. The same could be said of security. Production feedback will always speak louder than safety or security feedback, except on rare occasions. This makes the managerial control of safety/security extremely difficult.

To aid in this difficult task, both safety and security have developed techniques that attempt to assess the probability that system breaches will happen. Both disciplines assign values to the consequences of these events. In both fields, the expected frequency and severity of these events can be translated into an expected loss, and this value can be used in a cost/benefit analysis to select between protective interventions.

3.1.2 Attribution of failure

In both safety and security, defensive filters may be interposed between decision-makers and 'bad news.' As a result, safety or security problems may be blamed on operator carelessness or incompetence. This, in turn, may encourage management to respond with punitive actions towards staff, rather than address the underlying problem. This (unhelpful) position is

further consolidates by cataloguing the engineered safety devices/security mechanisms, and safe operating practices/security policies that have already been implemented.

3.2 Differences Between the Domains

It is sometimes argued that a major difference between these domains is that safety failures are frequently accidents, whereas security breaches are often deliberate (and so are likely to happen again and again). This difference is greatly reduced if we assume that the system exists in a dangerous world. When we focus on the victim/end-user (as a computer security policy must do) instead of the perpetrator/external cracker, we see that safety and security breaches *will* happen unless the victim takes appropriate steps to avoid them.

For example: crossing the street is dangerous whether or not drivers are trying to run you down. In both the safety (unintended collision) and security (assassination by car) versions of this scenario, the way to avoid the breach is to cross when there are no cars coming. In either version, ignoring the vulnerability (fast moving vehicles in your path) is likely to result in a serious injury. From this perspective, the two domains are similar and the basic etiologies of breaches are the same: committing unintended actions, or committing intended actions with the wrong goal (where the goal is not to cause an injury, or to cause a security breach).

The meaning that the society places upon *intended* breaches from inside the system however *is* different in some instances between safety and security. In most instances, violations of safety rules are not applauded. In many cases, violations of security rules are applauded. For example, there is a tradition of individuals releasing information about security flaws to the press either to gain publicity (the *publicity attack*- Schneier, 2000) or in the public interest. Sasse et al. (2001) report that being able to flaunt "petty" security regulations is a badge of seniority in many organizations.

4. A SPECIFIC SAFETY MODEL

In our view, Reason's *Generic Error Modeling System* (GEMS) is a comprehensive model for ensuring safety in organizations. It is a model that is informed by a detailed understanding of both individual (cognitive) and organizational (social) characteristics that direct user behavior (i.e. the Basic Elements of Production). We describe the model in two sections: firstly, failures at the individual (user) level, and secondly, failures at the system (organizational) level.

4.1 GEMS and Active Failures

The model posits three kinds of human error. Active failure, or active errors occur at the level of individuals (operators) in the system. In the security domain, the operators are end-users of computer systems in a large corporation. There are three error types:

- **slips** (attentional failures),
- **lapses** (memory failures) both of which are unintended actions that lead to a bad result; and
- **mistakes** (rule-based or knowledge-based mistakes) intended actions that lead to an unintended result.

Together with the traditional focus of computer security

- **violations** another intended action, however one that leads to a result that the user wishes, but other people do not

these form the class of unsafe acts (which in the domain of information security we re-label as ***insecure acts***).

4.2 Latent Failures, and Basic Elements of Production

For an accident (disaster, or security breach) to occur, *insecure acts* must combine with *latent failures* and or unusual environmental conditions. On their own, insecure acts are necessary, but not sufficient to cause system disasters. A consequence of this is that the model assumes that security breaches are due to people inside the organization, even if they are initiated by external agencies such as crackers, natural disasters, etc.

Latent failures can be thought of as something like resident pathogens. They are weaknesses built into the system, which predispose the system to disasters. Using the previous example of being run down by a car whilst crossing the road, the insecure act of not checking for oncoming traffic combines with the latent failure in the system of not having a pedestrian bridge to produce a disaster.

Latent failures act by promoting insecure acts and weakening the system's defenses. As with pathogen-related diseases, the catastrophic breakdown of complex, opaque technical installations requires the breaching of defenses by combinations of resident pathogens and sometimes bizarre local triggering events (Figure 2). Other things being equal, the system is likely to have more resident pathogens if it is more complex, interactive, tightly coupled, and opaque. However, while simpler systems are usually less interactive, less centralized, and more transparent, they tend to have fewer built-in defenses. As a result, relatively few pathogens can wreak greater havoc in simpler systems than in more advanced ones.

Having the concept of a *latent failure* as something that predisposes a system to security breaches necessitates some definition of system. A system or organization is described in the model in the following way (Figure 1.):

- **decision-makers** direct the organization at a strategic level (CEOs, VPs, etc.), and
- **line managers** implement the strategies. This implementation creates the
- **preconditions** (reliable equipment of the right kind, a skilled and motivated workforce, appropriate: attitudes, motivators, work schedules, maintenance programs, environmental conditions, codes of practice and policies, etc.) for
- **productive activities** which are the activities the organization carries out to attain its payoff, e.g. Internet service provision.
- **defenses** protect the organization, such as: uninterruptible power supplies, firewalls, virtual private networking, data backups, emergency generators, sprinkler systems, etc.

The causes of a disaster, or security breach, can be traced to failures at all levels listed in this model. To indicate this, the model has been relabeled appropriately (Figure 2).

A useful way for thinking about how deficiencies at a higher level are transformed into deficiencies at a lower level is as *failure types* converting into *failure tokens* (Hudson, 1988). For example, the line management deficiency of insufficient training is a pathogen type or *failure type* that manifests itself in the plane of *psychological precursors of insecure acts* (Figure 2) as a variety of *failure tokens*. This cascade effect (Figure 3) is the basis of the assertion that removing latent errors in higher parts of the system is more beneficial than removing errors further downstream (because these downstream errors are then prevented). This cascade effect can be included in a cost / benefit analysis to prioritize possible interventions.

4.3 Adapting the Model for Security

An immediate problem in the adaptation of this model to security is the precondition that there is "reliable equipment of the right kind". Commentators have suggested that such hardware and software does not exist in practice, and that economic realities make their development infeasible (e.g. Schneier, 2000). However, the reliance of much of information security on a *trusted computing base* is an acknowledged and as yet unresolved problem (cf. Baker, 1996).

We have given an outline of the model and provided some arguments why it might apply to security. How exactly can it be translated to the domain of security? How do we define different parts of the model? In particular, how do we decide which actions are *active* failures, and which are *latent*?

For example: most users do not look at the information about the login history of their UNIX account when they log in. Because of this, a user will be unable to detect when somebody used his account in the middle of the night. Is it an active failure? If the user's locus of attention was not on the information as it was presented, it could be argued that this behavior is due to a failure of attention. This is the definition of an active failure, therefore the action must be an active failure. However, the fact that the user regularly does not attend to the information when presumably he is meant to suggests that this is a violation. Commonplace violation of safety protocols to achieve greater efficiency is the definition of *routine violation*. However, not attending to the information that is displayed could be argued to be similar to not performing maintenance. It is an error that predisposes the system to disaster-by not attending to the information, the user reduces the probability that a cracker will be caught after compromising the system. The user has made the system more vulnerable, and this is the definition of a latent failure.

Because instances of activity or inactivity may be hard to classify, it would seem appropriate to apply every category to every instance of activity, and then decide at which part of the model to intervene. Though in theory it would be desirable to intervene against all possible interpretations of the causes of insecure actions, in practice it will be necessary for researchers to pick interventions based on budgets, time available, skills available, etc. The model guides us to consider the multifaceted and cascading nature of security problems so that we may better decide what should be done to solve them.

5. EVALUATION OF MODEL

We have given examples where traditional models of information security cannot explain or predict important security vulnerabilities or breaches. We have described the new model and its transfer to the domain of information security. In the sections below we will apply the new model to an example security breach, examine what the model gives us and also its advantages and disadvantages.

5.1 An Example of the Model In Use

We now return to one of our previous examples of a security breach-the theft of laptops, and our conjecture that most users will avoid use of encryption software. We will attempt to apply the new model to this example. This application will begin by considering each of the insecure acts and using them to label the chairman's behavior. We will then step through the other planes of the model (Figure 2) in turn, from psychological precursors of insecure acts, through line management deficiencies, to wrong decisions at board level.

Let us assume a company has policies that mandate the encryption of sensitive information such as mergers and acquisition (M&A) information – both when stored locally and transmitted by email. When the CEO, for instance, breaches this security rule, he commits a *violation*. Presumably, the CEO does not intend for the sensitive information to fall into the wrong hands, so we may rule out *sabotage*. It is more likely that the CEO did not use any encryption software as a matter of course. This is classed as a *routine violation*. The key component of insecure actions of this kind is motivation, particularly the search for efficiency. Unusable security mechanisms not only lead to insecure user behavior, but also affect user perception of the value and importance of security (Adams and Sasse, 1999). When public key encryption features are added, many users are unable to use them properly (Whitten and Tygar, 1999). However, here we deal only with the symmetric encryption of locally stored material, and assume that the Chairman is able to use this software successfully but is unwilling to do so.

Although conventional Human-Computer Interaction (HCI, or Human Factors Engineering) research may not directly tackle user motivation, it does so indirectly by reducing user workload and costs, and/or increasing task quality (Sasse et al., 2001). If we consider the task of using symmetric encryption software, we find that the greatest user workload is caused by user authentication. Let us assume that the CEO believes that encrypting files is too costly for him, and the workload of authentication is part of the reason for this. Each time a file is encrypted or decrypted, authentication must take place. When encrypting a file, some authentication token has to be applied so that the file can only be decrypted by authorized people (who are able to apply the same authentication key to the file).

In most encryption software, authentication is carried out via a password mechanism. The use of passwords is associated with several user costs: task completion is delayed by the time required to respond appropriately to the password dialog, and mental effort is required to recall the password. The effort required is particularly acute when the user is generating a password. Most security password policies (usually based on Federal Information Processing Guidelines - FIPS, 1985) mandate that each separate encryption should have a different password. This dramatically increases user workload. The

CEO is part of a system that predisposes him to breach security. Although passwords have many advantages as an authentication mechanism, their proliferation tends to reduce their effectiveness. This can be seen at help desks in Internet service providers and IT dependent organizations, where up to half of all Help-Desk calls are password-related (Murrer, 1999). This can become a significant financial burden.

There are other *authentication by knowledge* mechanisms, such as:

- Passfaces™ (http://www.idarts.com/)
- graphical passwords (Jermyn et al., 1999)
- pass sentences (Spector, 1994),
- pass algorithms (Haskett, 1984).

There are other paradigms of authentication using: what people possess (keys, *tokens*, smart cards, etc), what people are (*structural biometrics* such as hand geometry, fingerprints, retina scans, etc.) and what people do (*behavioral biometrics* such as signatures, voice prints, keystroke dynamics, etc). These authentication mechanisms will not be appropriate in many cases for reasons of economy, user acceptance, or task compatibility. However, let us assume that a risk assessment has been carried out which shows that the loss of information stored on a senior executive's laptop far outweighs the costs of implementing alternate security measures for such a small group of users. By selecting encryption software which uses passwords instead of less burdensome authentication mechanisms, a *latent error* is designed into the system.

What insight do we gain if we consider the chairman's actions to be a mistake? There are two types of mistake. The first type is the misapplication of a good rule. One such might be, *"I am the boss, therefore nobody will mess with my laptop"*. In the company HQ, this is a good rule that works well. Unfortunately, in the auditorium of a convention center, this rule can be considered to have been misapplied.

The second type of mistake is the application of *bad rules*. In this instance, the rule might have been something like *"I am in a public space in view of many people, therefore I will not be robbed"*. This is clearly a bad rule, because its contents may be generally considered to be wrong.

The next error type is the *lapse*; executives are busy people and moreover, a public presentation is an additional source of stress. In such conditions, the CEO could have been distracted at the moment he was going to encrypt the sensitive data on his laptop. This distraction caused him to momentarily lose track of what he was doing, and therefore miss out the step of clicking the encrypt button.

The final error type is the *slip*. Particularly under conditions of stress, but also more generally it is always possible to make some small slip in an action sequence. Intending to press the *encrypt* button, the CEO might have pressed the *sign* button, or selected another document that he had been working on and encrypting that instead of the intended one.

Stepping through the other planes of the model we first look at the psychological precursors of these insecure acts (plane two of Figure 1). The chairman believed that the costs of encrypting the contents of his laptop outweighed the risk of the laptop being stolen. The CEO was accustomed to being in a physically secure area. The chairman's priority was his performance at the podium at the conference, not the security of his laptop. Encryption software uses password authentication, instead of a mechanism with fewer user costs.

Turning now to *line management deficiencies* (plane 3), the manager in charge of arranging the visit did not hire a security guard for the chairman and his laptop. The security department had not audited the CEO's laptop, or installed appropriate encryption software or other resource denial functionality in it. The wrong decisions that led to these problems (plane 4) were probably the board seeing security as a financial burden, or a box to be ticked rather than an essential part of the business.

The paragraphs above give us an example of the use of the model, moving the finger of blame from the end user and pointing it throughout the organization. It also illustrates the cascade effect that problems higher up in the organization can have on the ground; error types being turned into error tokens.

5.2 What Does It Give Us?

Applying the model has helped to identify several different potential causes of or contributory factors to a security breach, and measures that could be taken to prevent them. Moreover, the model identifies hierarchies of causes, where elimination of particular vulnerabilities can remove several others further down the line.

Rather than being the main instigators of an accident, operators tend to be the inheritors of system defects created by poor design, incorrect installation, faulty maintenance and bad management decisions. To paraphrase Reason (1990), they usually only provide the final garnish to a lethal brew whose ingredients have already been long in the cooking.

10 years ago, the safety-critical systems community became aware that attempts to discover and neutralize these latent failures will have a greater beneficial effect upon system safety than more localized efforts to minimize active errors. The security community is now becoming aware that a view that is systematically wider than its traditional models is necessary. We suggest that a view of similar scope as the safety community's is necessary for security.

To date, much of the work of information security specialists has been directed at improving the secure transmission and storage of data. While this is undeniably an important enterprise, it only addresses a relatively small part of the total security problem, being aimed primarily at reducing the technical *preconditions* tip of the causal iceberg. One thing that has been profitably learned over the past few years is that, in regards to safety issues, the term "human factors" embraces a far wider range of individuals and activities than those associated with the frontline operation of the system. Indeed, a central theme of this work is that the more removed individuals are from these frontline activities (and, incidentally, from direct hazards), the greater is their potential danger to the system. We argue that this is also the case for security, and that the model presented here gives the security community a model of suitable scope.

5.3 Advantages

We have identified the following advantages of adapting the safety model to security:

1. It reminds us that there is more to security than software and mathematics; there are people too, and their interactions with the above and each other.
2. The model avoids premature formalization (cf. Dobson, 1993)
3. The model contains an *enterprise description*, which LaPadula (1993) has identified as a desirable property.
4. It situates the user in the context of the organization, rather than treating the user as a single unity devoid of context (Dobson, 1993).
5. Non-technical and social aspects such as organizational procedures and training are an essential part of the model, not an ancillary to it (cf. Dobson, 1993).
6. The model points to the area of security that is weakest, and where therefore the largest gains can be made.
7. The model gives the context for lower-level models such as BLP.
8. The model argues that information security should be given more status and resources.

5.4 Disadvantages

We currently see the following disadvantages:

1. The model depends on a trustworthy hardware and software foundation, in common with other models (cf. Baker, 1996).
2. The model is not easy to operationalise; the quality of its application depends on the expertise of the people who apply it within a particular organization. Again, this is also true for most other security models.
3. The model tells you which things are wrong, what appropriate goals are but not how to achieve them. This is similar to other models of security.
4. The model is currently under-specified: it does not list all of the important variables. However, this paper only represents a demonstration of the feasibility and first benefits of adapting the model to security – further work will be needed to generate a comprehensive model.

5.5 Further work

We see the following opportunities for further work:

- Integrating the model with existing security design and evaluation techniques.
- Applying the model to security issues other than user-authentication.
- Testing its validity for the design of new systems and for interventions in existing systems – is it better for design or redesign?
- Testing the scope of the model – is it helpful for security breaches at all levels of severity no matter how small, or only for disasters (which were the subject of the original model)?

6. SUMMARY

Reason's (1990) GEMS and the *basic elements of production* form a model that explained and predicted accidents in complex technical installations. We have demonstrated the feasibility of applying this model to the domain of information security. Our model is sufficiently general to encompass more traditional models of security such as BLP and CW, as well as their goals. We put two important classes of phenomena forward: active and latent failures. By focusing on latent failures in the system, we propose that security is better improved than by concentrating on the active failures of end-users. The model concentrates on the human components of work systems, which have been described as the weakest link in the security chain (Schneier, 2000). By focusing on the most important part of this area of weakness, it focuses on the area where the greatest gains can be made.

7. REFERENCES

[1] Adams, A. and Sasse, M.A. (1999), Users are not the enemy, *Communications of the ACM*, Vol. 42, No. 12. December, 1999.

[2] Anderson, R. (2001) *Security Engineering*. John Wiley and Sons; UK.

[3] Baker, D.B. (1996) Fortresses Built Upon Sand. in *Proceedings of the 1996 New Security Paradigms Workshop*.. Arrowhead, CA.: ACM Press https://www.acm.org/pubs/articles/proceedings/commsec/304851/p148-baker/p148-baker.pdf

[4] Bell, D., & LaPadula, L. (1973) *Secure Computer Systems: Mathematical Foundations and Model*, M74-244, MITRE Corp. Bedford, MA,

[5] Biba, K. (1977) *Integrity Constraints for Secure Computer Systems*. Tech. Rep. ESD-TR76-372, USAF Electronic Systems Division, Bedford, MA

[6] Brostoff, S. & Sasse, M. A. (2000): Are Passfaces more usable than passwords? A field trial investigation. In S. McDonald, Y. Waern & G. Cockton [Eds.]: People and Computers XIV - Usability or Else! Proceedings of HCI 2000 (September 5th - 8th, Sunderland, UK), pp. 405-424. Springer

[7] Bunnell, J., Podd, J., Henderson, R., Napier, R., & Kennedy-Moffat, J. (1997). Cognitive, associative and conventional passwords: Recall and guessing rates. *Computers and Security, 16*(7), 629-641.

[8] Clark, D. &. Wilson, D. (1987) A Comparison of Commercial and Military Computer Security Policies. In *Proceedings of the IEEE Symposium on Security and Privacy*, Oakland, CA.

[9] Dobson, J. (1993) new security paradigms: what other concepts do we need as well? *Proceedings of the 1993 workshop on new security paradigms*. August 225, 1993, Little Compton, RI. http://www.acm.org/pubs/contents/proceedings/commsec/283751/

[10] FIPS (1985) *Password Usage*. Federal Information Processing Standards Publication. May 30.

[11] Haskett, J. A. (1984). Pass-algorithms: a user validation scheme based on than knowledge of secret algorithms. *Communications of the ACM, 27*(8), 777-781.

[12] Hudson, P.T.W. (1988) Personal communication, In Reason, J. (1990) *Human Error*. Cambridge University Press. Cambridge, UK

[13] Jermyn, I., Mayer, A., Monrose, F., Reiter, M.K., Rubin, A.D. (1999) The Design and Analysis of Graphical Passwords. *Proceedings of the 8th USENIX Security Symposium*, August 23-36, 1999, Washington, D.C., USA

[14] LaPadula, J.. (1993) Prospect on security paradigms. *Proceedings of the 1993 workshop on new security paradigms*. August 225, 1993, Little Compton, RI. http://www.acm.org/pubs/contents/proceedings/commsec/283751/

[15] Lemos, R (2000) *Laptop thieves usually not after data.* http://www.zdnet.com/zdnn/stories/news/0,4586,2629471,00.html

[16] Lipson , D. A. and Fisher, H. F. (1999) Survivability-a new technical and business perspective on security. New security paradigms workshop. *Proceedings of the 1999 workshop on new security paradigms*, September 22 to 24, 1999, Caledonian hills Canada

[17] Murrer, E. (1999). Fingerprint Authentication. *Secure Computing*(March), 26-30.

[18] Menkus, B. (1988). Understanding the use of passwords. *Computers and Security, 7*(2), 132-136.

[19] Poulsen, K. (2000): *Mitnick to lawmakers: People, phones and weakest links.* http://www.politechbot.com/p-00969.html.

[20] Reason, J. (1990) *Human Error*. Cambridge University Press. Cambridge, UK

[21] Sasse, M. A., Brostoff, S. & Weirich, D. (2001), Transforming the 'weakest link': a human-computer interaction approach to usable and effective security. *BT Technical Journal*, 19 (3), 122-131. (Also at http://www.bt.com/bttj/)

[22] Schneier, B. (2000), *Secrets and Lies*, John Wiley & Sons, 2000.

[23] Spector, Y., & Ginzberg, J. (1994). Pass sentence - a new approach to computer code. *Computers and Security, 13*(2), 145-160.

[24] Whitten, A. & Tygar, J.D. (1999) Why Johnny can't encrypt: A usability evaluation of PGP 5.0, *Proceedings of the 8th USENIX Security Symposium*, August 1999, Washington.

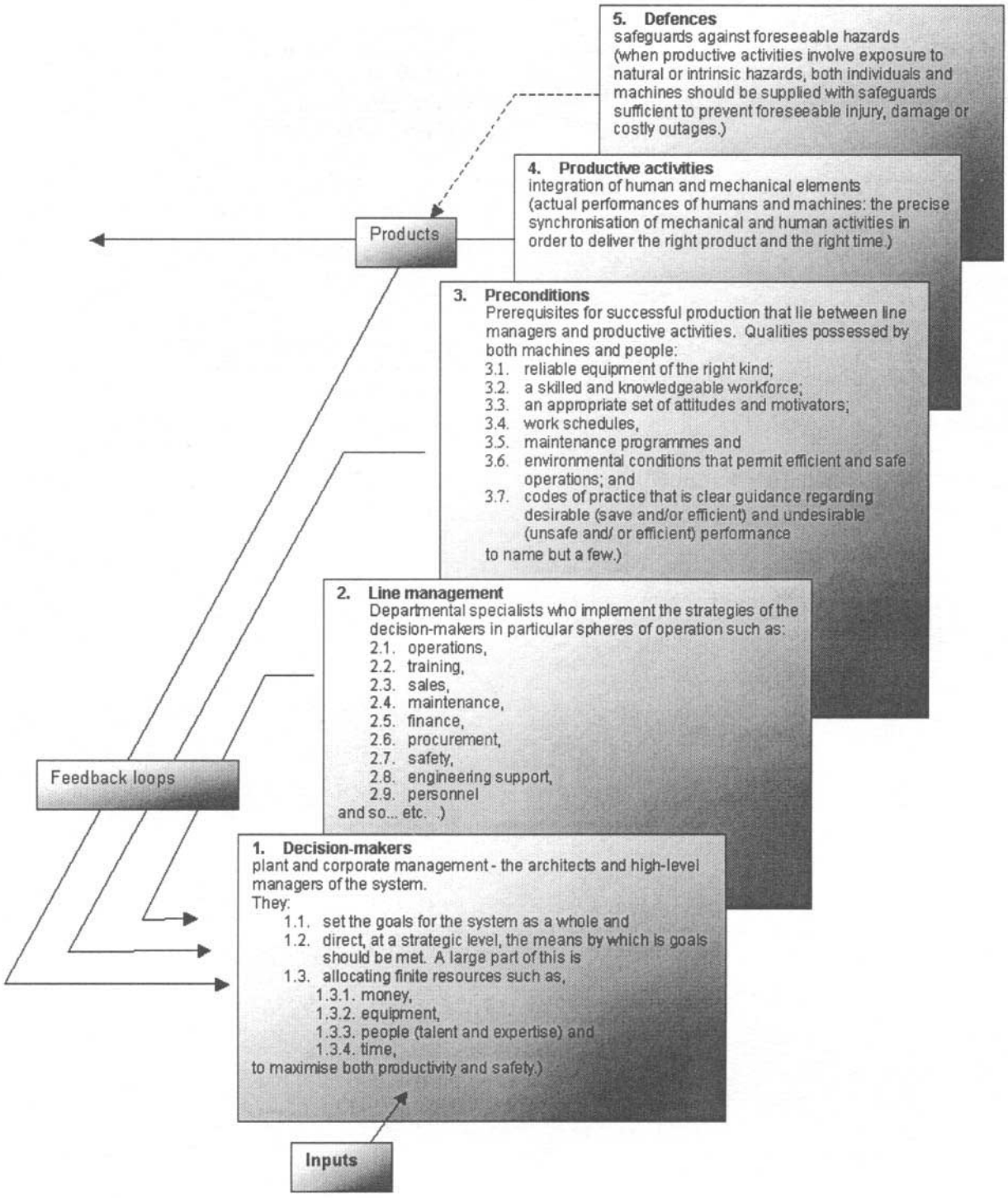

Figure 1 - GEMS and the Basic Elements of Production

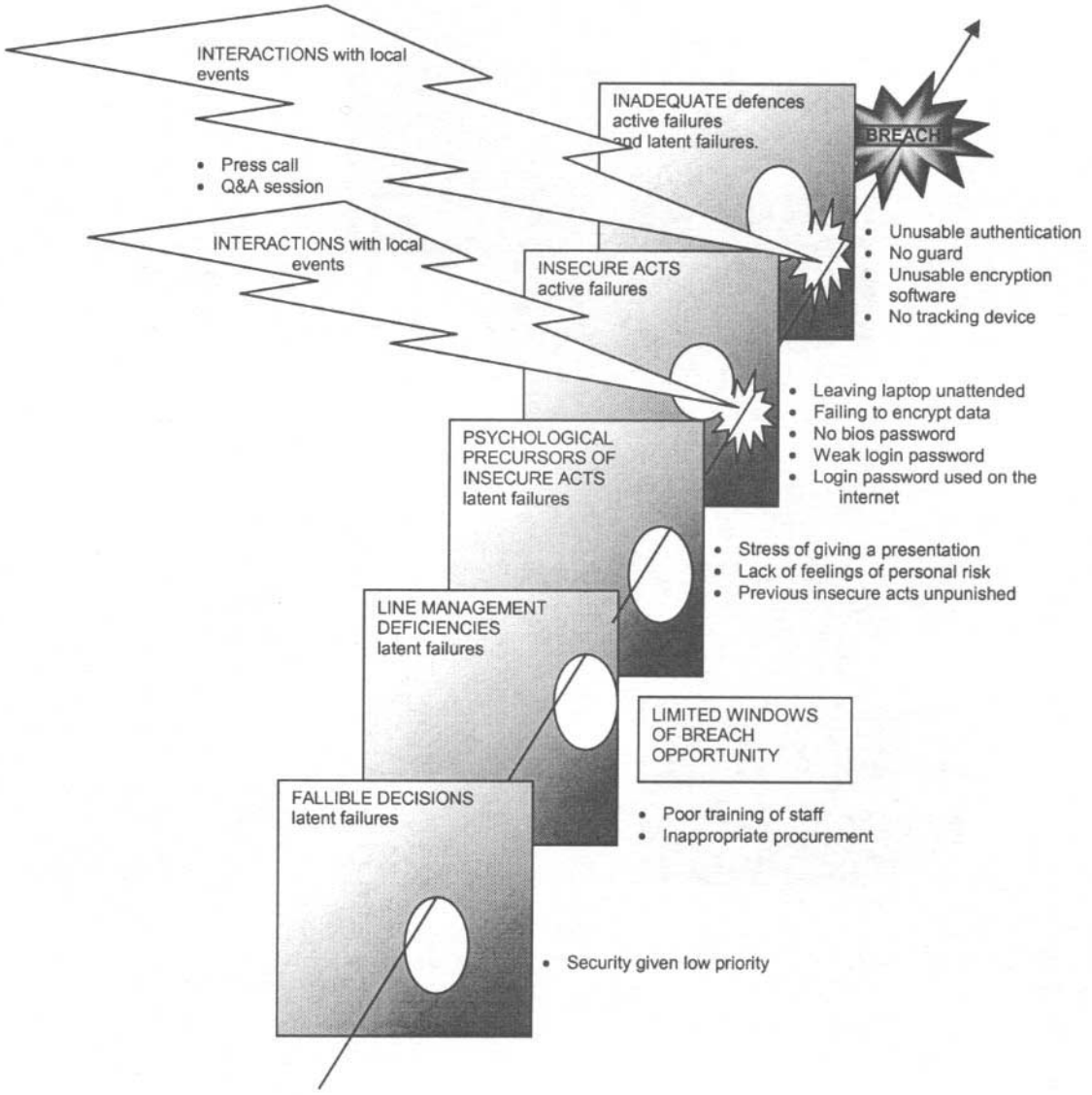

Figure 2 - errors at each of the elements of production, and the arrow of breach trajectory.

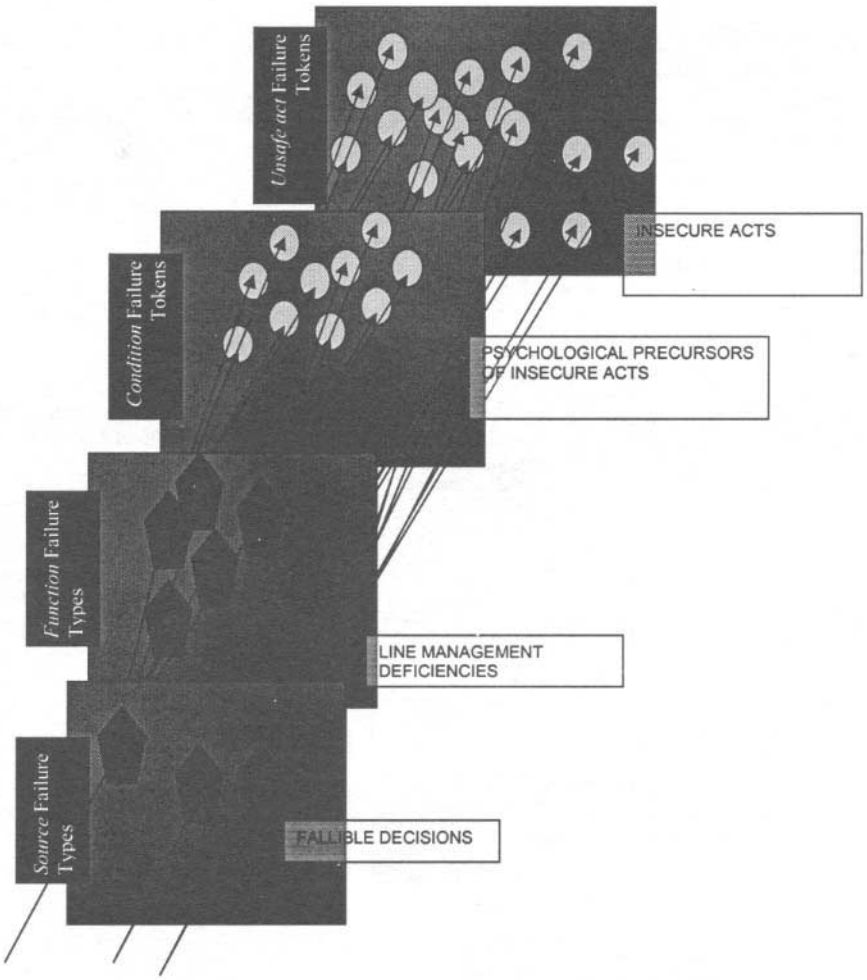

Figure 3 - The relationship between error types and error tokens