REASONING ABOUT SECURITY MODELS

John McLean

Computer Science and Systems Branch
Naval Research Laboratory
Washington, D.C. 20375

ABSTRACT

A method for evaluating security models is developed and applied to the model of Bell and LaPadula. The method shows the inadequacy of the Bell and LaPadula model, in particular, and the impossibility of any adequate definition of a secure system based solely on the notion of a secure state. The implications for the fruitfulness of seeking a global definition of a secure system and for the state of foundational research in computer security, in general, is discussed.

> And so of the virtues, however many and different they may be, they have all a common nature which makes them virtues; and on this he who would answer the question, 'What is virtue?' would do well to have his eye fixed.
>
> Plato, *Meno* (B. Jowett trans.) 72c6-d1

> For if you look at them you will not see something that is common to all, but similarities, relationships, and whole series of them at that...I can think of no better expression to characterize these similarities than "family resemblance"...
>
> Wittgenstein [1, §66-7]

> If a concept fundamental to a mighty science gives rise to difficulties, then it is surely an imperative task to investigate it more closely until those difficulties are overcome...
>
> Frege [2, p. II]

1. Introduction

Security is an especially hard property to prove rigorously about a program. It's not that proofs about security are intrinsically more difficult than proofs about other properties, but rather that the concept *security*, itself, is harder to explicate. For this reason, there has been a great deal of focus on rigorously defining the concept of security, or in the jargon of the trade, constructing formal security models. Such explications are important, for without them, many would regard it as impossible to establish in any meaningful way that a program is secure.

The security model developed by Bell and LaPadula [3] is the most widely accepted basis for verifying the security of systems [4]. It has been argued [5] that one reason developers should have confidence in the security provided by systems based on this model is a theorem, called the "Basic Security Theorem" (BST), proven about a formalization of the model by its authors [3, p. 90, corollary A1]. However, this confidence is misplaced since the BST can be proven for systems that directly contradict the notion of security embodied in the Bell-LaPadula model [6].

This paper presents a method for evaluating security models and applies the method to the Bell-LaPadula model. The results cast doubt on the Bell-LaPadula model and the fruitfulness of seeking global definitions of security. The existence of differing interpretations of the model cast doubt on the status of computer security's foundations in general.

2. How to Lend Credence to a Security Model

Current security models are formulated in terms of the concept of a *secure state*, *i. e.*, a definition that places restrictions on what a state can look like, a *secure transform*, *i. e.*, a definition that places restrictions on what a

Bell-LaPadula model's definition of *secure system*, it fails to satisfy the conditions required by our definition of a secure action. The fact that a definition of a secure system formulated in terms of our definition of a secure action is supposed to explicate the same concept as Bell and LaPadula's definition shows that either the former is too narrow or the latter is too wide.

The fact that system *Z* gives all subjects access to all objects shows that it is the Bell-LaPadula model that is inadequate. In fact, it should be clear that any explication of security based solely on the notion of a *secure state* must fail for a similar reason. At best such an explication can serve as a definition a *secure initial state*. The concept of a secure system must be explicated as one whose initial state is secure and whose system transform is secure.

5. The Bell and LaPadula Model Reconsidered

When presented with system *Z*, some have responded with an attitude of "Who cares?", while others have argued that the Bell-LaPadula model's explication of security consists of something more than the the model's definition of *secure system* and that this something more rules out systems such as *Z*.[6] With respect to the latter, the suggestion is that the model implicitly includes the tranquility principle, which prohibits changing the security level of an (active) object, or that it includes the particular Multics-based rules given in [3]. The first suggestion can easily be dismissed since the tranquility principle is clearly not part of the model as given in [3]. Not only is it not mentioned, it is violated by rule 11 of the Multics-based interpretation of the model. This is understandable since any model that did not permit violations of tranquility would be too confining to be practical.

The second suggestion can also be dismissed, but not as easily since [3] seems ambivalent with respect to it. Hence, we read that the rules are one of the model's three major facets [3, p. 5], yet that the the ss-, *-, and ds-properties constitute the "system characteristics that we desire to be maintained" [3, pp. 11-12] and that the rules are merely "one specific solution", a particular solution that "is in no sense unique, but has been specifically tailored for use with a Multics-based information system design."[3, p. 19] Though the rules are presented as being part of the model, the concept of a solution implies that they are not part of the model in any sense relevant to our considerations. If one explicates the concept of a Cartesian point's being five units from the origin by requiring that the point satisfies the equation $x^2+y^2=25$ and gives $(3,4)$ as a specific solution, we cannot conclude that the explication requires that any point (x,y) five units from the origin must have the property that $x+y=7$. Similarly, we cannot conclude from the particular solution Bell and LaPadula give, that a secure system must have any properties (beyond the properties of ss-, *-, and ds-security) that the particular solution has.

Rather, the particular system they specify serves as one example of a system that provably satisfies their definition of *secure system*. This is meant to justify our belief that the specified system is secure in some meaningful sense. Unfortunately, since system *Z* is another system that satisfies their definition of a secure system, the justification is unconvincing. Similarly, we cannot be sure that any system does not contain security flaws as serious, if not as obvious, as those of system *Z* simply because it satisfies the definition of *secure system* provided by the Bell-LaPadula model.

Perhaps the most compelling reason for believing that the Multics-based rules provide only an example of a secure system and not further properties a secure system must have is that no other reading of [3] makes sense of the relation between the definition of *secure system* and the Multics-based solution. The definition of a secure system and the particular solution don't convey the same set of constraints so it makes no sense to say that the two are different explications of *security*. Nor does it make sense to say that the rules are supposed to add

---

[6] All responses to system *Z* considered in this section are taken from *Computer Security Forum 5*, 18 (July 5, 1986), ed. Ted Lee for Arpanet distribution. System *Z* was originally presented in issue 14 (June 22, 1986) of the *Forum*, and additional responses appeared in issues 25 (September 23, 1986), 26 (October 5, 1986), 27-29 (all October 16, 1986), and 30-31 (all December 9, 1986).

modes in which an element of S can have access to an element of O.

Bell and LaPadula define a *system state* v as an element of $V=(B\times M\times F\times H)$, where

- B is the set of current accesses and is equal to $\mathcal{P}(S\times O\times A)$, with each of its elements denoted as b;
- M is the access permission matrix, where $A\supseteq M_{ij}$ is the set of access modes subject i may have to object j;
- F is a subset of $L^S\times L^S\times L^O$ where each $f\in F$ is a triple consisting of f_S, the security level (clearance) associated with each subject, f_O, the security level (classification) associated with each object, and f_C, the current security level for each subject, such that f_S dominates f_C; and
- H defines the current object hierarchy and is of no concern here.

The set of requests (*e. g.*, to acquire or rescind access to objects) is denoted by R, and the set of decisions (*e. g.*, *yes*, *no*, *error*) is denoted by D. $W\subseteq R\times D\times V\times V$ represents the actions of the system: (r,d,v_2,v_1) represents a request r yielding a decision d and moving the system from state v_1 to v_2. Letting T be the set of positive integers and X, Y, and Z the set of functions from T to R, D, and V, respectively, a *system* $\Sigma(R,D,W,z_0)$ is a subset of $X\times Y\times Z$ such that $(x,y,z)\in\Sigma(R,D,W,z_0)$ if and only if $(x_t,y_t,z_t,z_{t-1})\in W$ for each $t\in T$, where z_0 is the initial state of the system. Each triple $(x,y,z)\in\Sigma(R,D,W,z_0)$ is called an *appearance* of the system, and each quadruple (x_t,y_t,z_t,z_{t-1}) is called an *action* of the system.

The concept of a *secure state* is defined by three properties: the *simple security* (*ss-*) *property*, the **-property*, and the *discretionary security* (*ds-*) *property*. A state satisfies the ss-property if for each element of b that has an access mode of *read* or *write*, the clearance of the subject dominates (in the partial order) the classification of the object. A triple (s,o,x) satisfies the ss-property relative to f (rel f) if x is *execute* or *append*, or if x is *read* or *write* and $f_S(s)$ dominates $f_O(o)$.

A state satisfies the *-property if for each (s,o,x) in b, the current security level of s is equal to the classification of o if the access mode is *write*, dominates the classification of o if the access mode is *read*, and is dominated by the classification of o if the access mode is *append*. The concept of a triple satisfying the *-property rel f is analogous to satisfying the ss-property rel f. A state is said to satisfy the *-property relative to S', where $S'\subset S$, if this condition holds for all triples of b in which $s\in S'$. Subjects not in S' (and therefore not bound by the *-property relative to S') are called *trusted subjects*. It is worthwhile noting that since f_S dominates f_C the *-property implies the ss-property.

A state satisfies the ds-property if, for each member of b, the specified access mode is included in the access matrix entry for the corresponding subject-object pair. The concept of a triple satisfying the ds-property rel M is analogous to satisfying the ss-property rel f. A state is *secure* if and only if it satisfies the ss-property, the *-property relative to S', and the ds-property.

In addition to restricting subjects from having direct access to information for which they are not cleared, this concept of security is intended to prevent the unauthorized flow of information from a higher security level to a lower one. The *-property relative to S' specifically prevents nontrusted subjects from simultaneously having *read* access to information at one level and *write* access to information at a lower level.

Bell and LaPadula introduce analogous constraints on a system. A system appearance $(x,y,z)\in\Sigma(R,D,W,z_0)$ satisfies the ss-property if each state in the sequence $<z_0,z_1,\ldots>$ satisfies it. A system satisfies the ss-property if each of its appearances does. Analogous definitions introduce the notions of a system satisfying the *- and ds-properties and the concept of a *secure system*. Theorems A1, A2, and A3 (see below), for the ss-, *-, and ds-properties, respectively, show that a system $\Sigma(R,D,W,z_0)$ satisfies the property in question for any initial state that satisfies the property if and only if W (1) adds no new elements to b that would violate the property and (2) removes any elements that, following

the state change, would violate that property. The BST is presented without proof as a corollary of theorems A1, A2, and A3:

Basic Security Theorem: A system $\Sigma(R,D,W,z_0)$ is secure *iff* z_0 is a secure state and W satisfies the conditions of theorems A1, A2, and A3 for each action.[3]

Theorem A1: $\Sigma(R,D,W,z_0)$ satisfies the ss-property for any initial state z_0 that satisfies the ss-property *iff* W satisfies the following conditions for each action $(R_i,D_i,(b^*,M^*,f^*,H^*),(b,M,f,H))$:

(i) each $(s,o,x)\in b^*\sim b$ satisfies the ss-property rel f^*;

(ii) if $(s,o,x)\in b$ does not satisfy the ss-property rel f^*, then $(s,o,x)\notin b^*$.

Theorem A2: $\Sigma(R,D,W,z_0)$ satisfies the *-property relative to S' for any initial state z_0 that satisfies the *-property relative to S' *iff* W satisfies the following conditions for each action $(R_i,D_i,(b^*,M^*,f^*,H^*),(b,M,f,H))$:

(i) for each $s\in S'$, any $(s,o,x)\in b^*\sim b$ satisfies the *-property with respect to f^*;

(ii) for each $s\in S'$, if $(s,o,x)\in b$ does not satisfy the *-property with respect to f^*, then $(s,o,x)\notin b^*$.

Theorem A3: $\Sigma(R,D,W,z_0)$ satisfies the ds-property *iff* the initial state z_0 satisfies the ds-property and W satisfies the following condition for each action $(R_i,D_i,(b^*,M^*,f^*,H^*),(b,M,f,H))$:

(i) if $(s_k,o_l,x)\in b\ ^*\sim b$, then $x\in M^*_{kl}$;

(ii) if $(s_k,o_l,x)\in b$ and $x\notin M^*_{kl}$, then $(s_k,o_l,x)\notin b^*$.

On the face of it the BST looks like what we want, *i. e.*, an alternative formulation of security given in terms of state transitions that we can compare to the Bell-LaPadula model, but it's not. The reason can be seen by examining A1-A3: the concept of a secure action (transform) is defined solely in terms of a secure state. A transform can alter b, f, or M if the resulting state does not violate security. Put more baldly, a transform is defined to be secure if it leads to a secure state. The trouble with the theorem as it stands is that if our definition of *secure state* is wrong, our theorem is unaffected. In fact it has been shown in [6] that the BST holds for any system as long as its state sequence is indexed in a way that supports induction. The system can permit subjects to read up, write down, or whatever. What we need to justify the Bell-LaPadula model is an independent definition of security we can use to validate our definition of a secure state.

4. A Reformulation of the Bell and LaPadula Model

In light of the inadequacy of the BST to justify the Bell-LaPadula model, we must develop an independent definition of *secure transform* or, in Bell and LaPadula's terminology, of *secure action*. To this end, consider the following definitions:

Definition: An action $(R_i,D_i,(b^*,M^*,f^*,H^*),(b,M,f,H))$ is *ss-secure iff*

(i) if $(s,r,o)\in b^*\sim b$ or $(s,w,o)\in b^*\sim b$, then $f_S(s)$ dominates $f_O(o)$, and $(M^*,f^*,H^*)=(M,f,H)$;

(ii) if $f_S(s)\neq f^*_S(s)$, then (i) b does not contain any triples of the form (s,r,o) or (s,w,o) where $f_O(o)$ is not dominated by $f^*_S(s)$, and (ii) $f^*_O=f_O, f^*_C=f_C$, and $(b^*,M^*,H^*)=(b,M,H)$;

(iii) if $f_O(o)\neq f^*_O(o)$, then (i) b does not contain any triples of the form (s,r,o) or (s,w,o) where $f^*_O(o)$ is not dominated by $f_S(s)$ and (ii) $f^*_S=f_S, f^*_C=f_C$, and $(b^*,M^*,H^*)=(b,M,H)$.

Definition: An action $(R_i,D_i,(b^*,M^*,f^*,H^*),(b,M,f,H))$ is **-secure iff*[4]

(i) if $(s,r,o)\in b^*\sim b$ $[(s,w,o)\in b^*\sim b$, $(s,a,o)\in b^*\sim b]$, then $f_C(s)$ dominates $f_O(o)$ $[f_O(o)=f_C(s), f_O(o)$ dominates $f_C(s)]$, and $(M^*,f^*,H^*)=(M,f,H)$;

---

[3] In [3] an appearance satisfies the ss-property if each state in $<z_1,z_2,\ldots>$ satisfies the property; no restriction is placed on z_0. Nevertheless, the intent is clear since without this restriction, the BST as stated in [3] is false. See [6].

[4] For simplicity, we assume that no subjects are trusted, *i. e.*, that $S'=S$.

(ii) if $f_C(s) \neq f^*_C(s)$, then (1) b does not contain any triples of the form (s,r,o) $[(s,w,o), (s,a,o)]$ where $f_O(o)$ is not dominated $f^*_C(s)$ $[f_O(o) \neq f^*_C(s), f^*_C(s)$ is not dominated by $f_O(o)]$, and (2) $f^*_O = f_O$, $f^*_S = f_S$, and $(b^*,M^*,H^*)=(b,M,H)$;

(iii) if $f_O(o) \neq f^*_O(o)$, then (1) b does not contain any triples of the form (s,r,o) $[(s,w,o), (s,a,o)]$ where $f^*_O(o)$ is not dominated $f_C(s)$ $[f^*_O(o) \neq f_C(s), f_C(s)$ is not dominated by $f^*_O(o)]$ and (2) $f^*_C = f_C$, $f^*_S = f_S$, and $(b^*,M^*,H^*)=(b,M,H)$.

Definition: An action $(R_i,D_i,(b^*,M^*,f^*,H^*),(b,M,f,H))$ is *ds-secure iff*

(i) if $(s_x,\phi,o_y) \in b^* \sim b$, then $\phi \in M_{xy}$ and $(M^*,f^*,H^*)=(M,f,H)$;

(ii) if $\phi \in M_{xy} \sim M^*_{xy}$, then $(f^*,H^*)=(f,H)$ and $\{(s_x,\phi,o_y)\} \in b \sim b^*$;

(iii) if $\phi \in M_{xy} \sim M^*_{xy}$ or $\phi \in M^*_{xy} \sim M_{xy}$, then the subject executing R_i owns o_y and $(b^*,f^*,H^*)=(b,f,H)$.[5]

Definition: An action $(R_i,D_i,(b^*,M^*,f^*,H^*),(b,M,f,H))$ is *secure iff* it is ss-secure, *-secure, and ds-secure.

It is worthwhile examining this definition in detail. On the face of it, the set of secure actions is exactly the set of actions that meet the conditions of the BST. This is partly correct in that a secure action always takes one secure state to another, as is proven in the following theorem.

Theorem: A system $\Sigma(R,D,W,z_0)$ is secure if z_0 is a secure state and each action $(R_i,D_i,(b^*,M^*,f^*,H^*),(b,M,f,H)) \in W$ is secure.

Proof: We prove the theorem by induction. Since z_0 is secure by hypothesis, we can limit ourselves to the case where z_n is secure and show that z_{n+1} must be secure. We show this by proving that if W consists entirely of secure actions and if z_n is secure, then any action in W applied to z_n satisfies the conditions of the BST. Since, as noted in Section 3 above, the *-property implies the ss-property, A2 implies A1 so we only have to consider A2 and A3. For A2 to be false, there must be a $(s,o,x) \in b^*$ that fails to satisfy the *-property rel f^*. Since z_n is secure by hypothesis, either (s,o,x) is a new access or f was changed by W so as to violate *-security. The latter is impossible since by clauses (ii) and (iii) of the definition of a *-secure action f can only be so altered if $b^* = b$ and b is *-secure relative to f^*. Alternatively, if (s,o,x) was added by W, clause (i) of the definition of a *-secure action guarantees that (s,o,x) is *-secure rel $f=f^*$, and hence, that A2 is true. For A3 to be false, there must be a $(s_k,o_l,x) \in b^*$ such that $x \notin M^*_{kl}$. Since z_n is secure by hypothesis, either $b^* \neq b$ or $M^*_{kl} \neq M_{kl}$. If the former, then clause (i) of the definition of a ds-secure action guarantees that the added access is secure relative to $M^* = M$, and hence, that A3 is true. If the latter, then an access must have been dropped from M. But clause (ii) of the definition of a ds-secure action guarantees that this same access must have been dropped from b so A3 is again true, and we are done.∎

Hence, secure actions applied to a secure state lead to a secure state, and in this respect, our definitions mirror the BST. However, although our definition of a secure action satisfies the *if*-clause of the BST, it fails the *only if*-clause. It's not the case that any action that takes a system from one secure state to another secure state is secure. As an example, consider the system Z whose initial state is secure and that has only one type of action:

> When a subject s requests any type of access to an object o, every subject and object in the system is downgraded to the lowest possible level, permission is entered into the access matrix M, and the access is recorded in the current access set b.

It is easy to see that system Z's actions always leads to a secure state (in the Bell-LaPadula sense) and hence that system Z is certifiably secure by the lights of the Bell-LaPadula model. But though system Z satisfies the BST and the

---

[5]There is really no analogue to this condition in the Bell-LaPadula axioms, but it seems an intuitive requirement. Nothing in this paper depends on secure actions having this property.

Bell-LaPadula model's definition of *secure system*, it fails to satisfy the conditions required by our definition of a secure action. The fact that a definition of a secure system formulated in terms of our definition of a secure action is supposed to explicate the same concept as Bell and LaPadula's definition shows that either the former is too narrow or the latter is too wide.

The fact that system *Z* gives all subjects access to all objects shows that it is the Bell-LaPadula model that is inadequate. In fact, it should be clear that any explication of security based solely on the notion of a *secure state* must fail for a similar reason. At best such an explication can serve as a definition a *secure initial state*. The concept of a secure system must be explicated as one whose initial state is secure and whose system transform is secure.

5. The Bell and LaPadula Model Reconsidered

When presented with system *Z*, some have responded with an attitude of "Who cares?", while others have argued that the Bell-LaPadula model's explication of security consists of something more than the the model's definition of *secure system* and that this something more rules out systems such as *Z*.[6] With respect to the latter, the suggestion is that the model implicitly includes the tranquility principle, which prohibits changing the security level of an (active) object, or that it includes the particular Multics-based rules given in [3]. The first suggestion can easily be dismissed since the tranquility principle is clearly not part of the model as given in [3]. Not only is it not mentioned, it is violated by rule 11 of the Multics-based interpretation of the model. This is understandable since any model that did not permit violations of tranquility would be too confining to be practical.

The second suggestion can also be dismissed, but not as easily since [3] seems ambivalent with respect to it. Hence, we read that the rules are one of the model's three major facets [3, p. 5], yet that the the ss-, *-, and ds-properties constitute the "system characteristics that we desire to be maintained" [3, pp. 11-12] and that the rules are merely "one specific solution", a particular solution that "is in no sense unique, but has been specifically tailored for use with a Multics-based information system design."[3, p. 19] Though the rules are presented as being part of the model, the concept of a solution implies that they are not part of the model in any sense relevant to our considerations. If one explicates the concept of a Cartesian point's being five units from the origin by requiring that the point satisfies the equation $x^2+y^2=25$ and gives $(3,4)$ as a specific solution, we cannot conclude that the explication requires that any point (x,y) five units from the origin must have the property that $x+y=7$. Similarly, we cannot conclude from the particular solution Bell and LaPadula give, that a secure system must have any properties (beyond the properties of ss-, *-, and ds-security) that the particular solution has.

Rather, the particular system they specify serves as one example of a system that provably satisfies their definition of *secure system*. This is meant to justify our belief that the specified system is secure in some meaningful sense. Unfortunately, since system *Z* is another system that satisfies their definition of a secure system, the justification is unconvincing. Similarly, we cannot be sure that any system does not contain security flaws as serious, if not as obvious, as those of system *Z* simply because it satisfies the definition of *secure system* provided by the Bell-LaPadula model.

Perhaps the most compelling reason for believing that the Multics-based rules provide only an example of a secure system and not further properties a secure system must have is that no other reading of [3] makes sense of the relation between the definition of *secure system* and the Multics-based solution. The definition of a secure system and the particular solution don't convey the same set of constraints so it makes no sense to say that the two are different explications of *security*. Nor does it make sense to say that the rules are supposed to add

---

[6] All responses to system *Z* considered in this section are taken from *Computer Security Forum 5*, 18 (July 5, 1986), ed. Ted Lee for Arpanet distribution. System *Z* was originally presented in issue 14 (June 22, 1986) of the *Forum*, and additional responses appeared in issues 25 (September 23, 1986), 26 (October 5, 1986), 27-29 (all October 16, 1986), and 30-31 (all December 9, 1986).

additional constraints that a secure system must meet. For one thing, their Multics orientation makes them too restrictive to serve this purpose [3, pp. 20-25], and for another, on this interpretation the Bell-LaPadula definition of *secure system* would serve no purpose. It would be redundant since any system that meets the conditions implicit in the rules satisfies the definition. Finally, this interpretation does not do justice to the text. If the rules were to be included in the concept of being a secure system, then the definition of such a system would say that it must satisfy the ss-, *-, and ds-security properties and the rules, the BST would have to include the rules, *etc.*

The only alternative is our view that the Bell-LaPadula definition of *secure system* is supposed to provide all the security-relevant constraints such a system must meet. And though system *Z* shows that this view is untenable, it is, in fact, the only option that makes sense. Those who accept this view yet are still complacent about system *Z* seem to view the Bell-LaPadula model as only a framework for representing systems, rather than as a criterion that secure systems should conform to. In this view showing that a system conforms to the model says nothing about whether the system is secure. Ignoring the question of why we need such a complicated framework for modeling systems and the question of whether this claim makes sense in light of the prominent role played by the definition of *secure system* in the model, we can still say that this view certainly runs counter to the way the model is generally regarded by the computer security community.[7] If nothing else, the fact that there can be so much disagreement over something so established and so fundamental is sufficient to cause concern and provide ample reason for dismissing a response of "Who cares?".

6. Foundations for Computer Security

Several comments are in order. First, as noted above, the definitions of secure actions are more restrictive than what is required by the Bell-LaPadula model. Some of them could change without violating security. For example, part (*ii*) of the definition for *ds-security* could prohibit subjects from removing permissions to their files if it meant removing a current access. However, though such a change alters the flavor of our concept of security, it does not yield a strikingly different one. A more significant change would be to follow [9] and introduce a *system security officer* and the concept of a *role*, such as *downgrader*. Such possible changes may be necessary (see below) and, at the least, raise the question of why we prefer one formulation over another.

Second, we must decide where the original Bell-LaPadula model fits in. For a system to be secure, its actions must be secure by the definitions given above, and its initial state must meet the definition of a secure state given by Bell and LaPadula. However, this leaves us with a hybrid definition of security and not two separate definitions we can compare. Further, it should be clear that no explication of security can be based solely on the notion of a secure transition. The concept of a secure initial state is always required.

The last statement is the rub. System *Z* shows that no adequate explication of security can be based solely on the notion of a secure state, and we have just seen that there can be no adequate explication based solely on the notion of a secure transition. Hence, our original plan of comparing two explications of security, though successful in showing an inadequacy in the Bell-LaPadula Model, ultimately fails. Our hybrid approach may be adequate, but we have no alternative explication to compare with it. We can appeal to intuition, but such appeals are insufficient, especially in light of weaknesses displayed in the intuitively correct Bell-LaPadula Model. In fact, since neither model has a system security officer, our reformulation shares with the original model what, to my taste, is an all too cavalier approach to altering f_s and f_o. The ability to raise a subject's f_s as

---

[7] See, *e. g.*, [8, pp. 64-65, 89, 111] which all but requires that a formal security policy model used for formal design verification be state-based à la the Bell-LaPadula model, and which states both that such design verification "can effectively protect classified or other sensitive information stored or processed by the system" and that "the *-property is sufficient to prevent the compromise of information by Trojan Horse attacks." System *Z* shows that both claims are false.

long as it has no current accesses or lower an object's f_o as long as no subject is currently accessing it can obviously lead to security breeches. Even the ability to alter f_c is unsettling. If processes have no memory, then the *-property is too restrictive since there is no need to prohibit a write down as long as nothing is *concurrently* being read on a higher level. If processes have memory, freely lowering f_c obviously presents problems.[8]

The moral may be that we should change tactics. Instead of searching for some Platonic form of security, it may be time to realize that there are several concepts of security that bear only, to use Wittgenstein's phrase, a family resemblance to each other. If this is correct, our task should be to look at each application separately where our intuitions are more reliable and explicate the concept of security relevant to it [9].

In any event, it is certainly time for the computer security community to begin a thorough examination of our foundations. The Bell-LaPadula model was a monumental piece of work, but it has lived in an overly sheltered environment which has permitted it to survive beyond its rightful time. Like a pampered offspring, it has endured, not because it is fit, but because it has been protected from harm.

As it is presented in [3], the model is inadequate to bear the weight the computer security community has placed on it, and those who insist on its soundness have conflicting views of it which are inconsistent with [3]. Hence, we have developed an environment where our documented foundations are inadequate, yet shielded from adversity by appeals to implicit assumptions "which everybody knows about" (even if people disagree on what these assumptions are!). Such an environment prevents the examination of the foundations that actually underlie our systems and will eventually impede the development of new systems. Until the implicit foundations many in the computer security community claim to exist are documented and subjected to critical scrutiny, our faith in our systems will be unjustified. Perhaps worse, we will be doomed to a cycle where as practitioners retire, the assumptions that "everybody knows" will be forgotten, leaving only the information contained in the false publications, and then rediscovered as our new systems fail, only to be forgotten again. Such is the path to neither science nor security.

[8] I first heard this point from Debbie Cooper. The only interpretation of the *-property I can think of that makes sense is if we assume that processes can remember things, but only until their current security level changes. Even then, the property should only prohibit writing to a lower level than a previous read.

Acknowledgments

I wish to thank Carl Landwehr for his comments on an earlier draft of this paper and all those who contributed their thoughts on system *Z* to *Computer Security Forum 5*. Of the latter, I especially wish to thank Don Good for some much appreciated encouragement and for what I learned from his insightful contribution to issue 27.

References

[1] L. Wittgenstein, *Philosophical Investigations* (translated by G. E. M. Anscombe), The Macmillan Company, New York, 1953.

[2] G. Frege, *The Foundations of Arithmetic* (translated by J. L. Austin), Northwestern University Press, Evanston, 1968.

[3] D. E. Bell and L. J. LaPadula, Secure computer system: unified exposition and Multics interpretation, MITRE MTR-2997, Mar. 1976. Available as NTIS AD-A023 588.

[4] C. E. Landwehr, Best available technologies for computer security, *IEEE Computer*, Jul. 1983.

[5] Panel session on Bell-LaPadula and alternative models of security, S. B. Lipner moderator, IEEE Symposium on Security and Privacy, Apr. 1983.

[6] J. McLean, A comment on the "Basic Security Theorem" of Bell and LaPadula, *Information Processing Letters 20* (1985), 67-70.

[7] J. McLean, C. E. Landwehr, and C. L. Heitmeyer, A formal statement of the MMS security model, *Proc. 1984 Symposium on Security and Privacy*, IEEE Computer Society Press, 1984.

[8] *Department of Defense Trusted Computer System Evaluation Criteria*, CSC-STD-001-83, National Computer Security Center, Ft. Meade, MD, Aug. 1983.

[9] C. E. Landwehr, C. L. Heitmeyer, and J. McLean, A security model for military message systems, *Transactions on Computer Systems 2*, 3 (Aug. 1984), 198-222.