6 INTEGRITY AND DENIAL OF SERVICE

Does one's integrity ever lie in what he is not able to do?
Flannery O'Connor,
Wise Blood (foreword),
1952.

In silence man can most readily preserve his integrity.
Meister Eckhart,
Directions for the Contemplative Life,
c. 1320.

6.1. INTRODUCTION

Most, if not all, of the existing published work in composition of security properties focuses on the relatively well-understood and well-formalized domain of mandatory non-disclosure. Perhaps this can be partially explained as a simple continuation of the security community's habit of emphasizing non-disclosure, but this is likely not the entire story. The pre-existing body of formal models and definitions of non-disclosure supplied the proper foundation on which theories of composition could be built. Several of these were examined in detail earlier. The number and variety of approaches is surprising, given the infancy of the subject.

Formal definitions and models of other aspects of security are either rudimentary and not widely accepted, or are altogether lacking. Again, this is partly a result of an historical preoccupation, but is also reflective of real difficulties inherent in the formal treatment of these other security concerns.

Informal treatments are somewhat easier to come by. One reasonable, though certainly not universally acceptable, intuitive understanding of immunity from denial of service attacks can be phrased as ''the system will continue to function despite the actions of unauthorized individuals.'' By the phrase ''continue to function'', we mean that the system will function in as correctly and as timely a manner in the presence of interfering activity as without. In this informal setting, use of the words ''as correctly'' enables us to rather easily decouple denial of service from other system faults that might cause incorrect operation.

For example, if an untrusted application program contains a bug, then the system when running that program will not meet its operational objectives and so cannot be said to be operating correctly. However, in the sense that this is a ''legitimate'' bug (an awkward phrase, but the only one that comes to mind), this is not a denial of service problem as defined above, and perhaps not really a security problem of any stripe.

But, if the application did not have the bug yesterday and the system permitted its introduction by someone not authorized to modify production software, then this might be classed as a denial of service. It could, of course, also be called other things, but if we accept the view that the production software is a system resource that has been rendered unusable, the denial of service label does seem to fit.

This distinction can be made informally with relative ease, but the discipline of formal specification does not permit such shortcuts. A precise definition of ''denial of service'' is difficult to achieve without a similarly meticulous definition of exactly what ''service'' is.

This difficulty is shared by other security attributes, and is one of the fundamental ways in which they differ from non-disclosure. Nondisclosure is entirely a *safety* property — it makes no demands on what the system must do, but does does say what it must not do. Furthermore, nondisclosure is readily defined and enforced at the application boundary. The application's internal function is unimportant, as is the intended meaning of its inputs and outputs, because nondisclosure can be defined as the lack of correlation between uninterpreted message streams.

This definition process can also be carried out at the subsystem level, where we again find that the nondisclosure properties of a subsystem can be defined in terms of the patterns of its inputs and outputs without reference to their meaning. Knowledge of the component function is eventually necessary in order to represent that particular component in the model, otherwise we cannot characterize it well enough to measure it against the nondisclosure property definition, but this knowledge is not necessary when dealing with the property in the abstract.

This characteristic greatly simplifies the nondisclosure composition problem. The security of the composite system can be defined without regard to its intended operation, and without regard to the interpretation of the signals exchanged by the components. Thus, composable non-disclosure can be defined without requiring that the components work together in a correct way, or even in a meaningful way.

The same cannot be said of other security properties. Integrity, denial of service protection, audit, authentication, etc. all imply certain correct function of a system, and cannot be suitably defined for a ''black-box'' system with uninterpreted input and output ports. For this reason, techniques developed in modelling nondisclosure and in investigating its composability are not necessarily directly transferrable to other security areas.

In this report, we will consider several types of models and specification systems proposed for various security properties. Each model will be evaluated with respect to its composability properties, and also for any characteristics that might induce a meaningful order among systems evaluated against the model. Our objective here is not to exhaustively enumerate all possible models, but rather to obtain a cross section of security modelling techniques that we hope will be representative of the breadth of models that may be proposed elsewhere in the future.

6.2. THE YU-GLIGOR DENIAL OF SERVICE MODEL