Defining, Computing and Interpreting Trust

Panel Moderator: Daniel Faigin, Engineering Specialist, The Aerospace Corporation
Panelists: Michael Clifford, Information Security Analyst, The Aerospace Corporation
Matt Bishop, Associate Professor, The University of California at Davis
Marshall Abrams, Principal Scientist, The MITRE Corporation

Panel Theme

Very little agreement exists in the security community (or even outside of it) as to what trust actually means, and how to go about computing it. Various trust models use transitive, multilevel, hierarchical or relativistic methods of handling trust. The problem can be broken into three parts: how trust is defined, how an assertion of trust should be interpreted, and how trust relationships, or assertions of trust can be efficiently and correctly modeled and computed. For example, should trust be defined in terms of a mechanistic process, such as an evaluation against baseline criteria, as a deductive process based upon axioms, or as a subjective and interpretive process in which the meaning of trust is in constant flux? Or should some other method of determining trust be used? Once a trust relationship is asserted, should you accept or ignore the assertion, or use it to modify your own beliefs? Do you trust another entity to make such an assertion at all? If trust is defined and interpreted non-uniformly, can it be computed at all? The panelists will offer three different perspectives on how trust should be defined, computed and interpreted.

Position Statements

Marshall Abrams

Asking for a general definition, or attitude toward, trust is much too broad. There is a simple definition that includes a trap. That definition is that trust in an IS means that I believe that it will do what I expect it to do and will not do what I don't expect it to do. The trap is that if I expect it to fail to protect all valuable information, and it is easily compromised, then it is doing what was expected and should therefore be considered trusted. I would say that trusting an IS means that I rely on that IS to enforce the policy concerning protecting the assets entrusted to it. The policy defines what is acceptable and unacceptable usage. Trust may be based on many factors, including development process, pedigree of the developers, testing, legal remedies, and necessity. Sometime trust is based on extensive testing or documentation; sometime there is blind trust with no basis at all.

Matt Bishop

"Trust" is "an axiomatic acceptance of some quality or statement." For example, if I trust my system not to leak information, I believe that it won't transfer information covertly. I may have no basis for that belief, but I accept it because I trust my system. Trust can be derived; I can say I trust something (call it B) if I trust X, Y, and Z, and I can prove B follows from those. The same applies for interpretation of trust. If I accept a model of a system (or of any phenomenon, for that matter), I trust its assumptions to be correct (or acceptable to me). Otherwise I have a false hypothesis, and any conclusions derived may, or may not, be true.

Michael Clifford

I believe that trust is a concept which lends itself very well to an egocentric perspective of the world, but very poorly to hierarchical or transitive perspectives, where trust from one entity is imposed upon another. I think that trust should be dynamically defined by the user on a case by case basis. Interpretation of trust should also be dynamic and interpretive. Definitions and interpretations which work in one case may not work in another. Trust models which do not have these properties work only in limited domains. Accurate computation of trust with traditional models is not possible, because these models do not reflect the views of the user. However, trust models which are dynamic, interpretive and egocentric would not only solve this problem, but would also be universally applicable and computationally feasible.