

# Extended Abstract: Complexity and Vulnerability Analysis<sup>1</sup>

Stephen F. Bush

[bushsf@research.ge.com](mailto:bushsf@research.ge.com)

<http://www.research.ge.com/~bushsf>

## I. INTRODUCTION

The vulnerability analysis technique presented in this paper takes into account the innovation of an attacker. A metric for innovation is not new; 700 years ago William of Occam suggested a technique [6]. The salient point of Occam's Razor and complexity-based vulnerability analysis is that the better one understands a phenomenon, the more concisely the phenomenon can be described. This is the essence of the goal of science: to develop theories that require minimum size to be fully described. Ideally, all the knowledge required to describe a phenomenon can be algorithmically contained in formulae; formulae that are larger than necessary lack a full understanding of the phenomenon. Consider an attacker as a scientist trying to learn more about his environment, that is, the target system. Parasitic computing [1] is a literal example of a scientist studying the operation of a communication network and utilizing it to his advantage in an unintended manner. The attacker as scientist generates hypotheses and theorems. Theorems are attempts to increase understanding of a system by assigning a cause to an event, rather than assuming all events are randomly generated. If theorem  $x$ , described in bits, is of length  $l(x)$ , then a theorem of length  $l(m)$ , where  $l(m)$  is much less than  $l(x)$ , is not only much more compact, but also  $2^{l(x)-l(m)}$  times more likely to be the actual cause than pure chance [6]. Thus, the more compactly a theorem can be stated, the more likely one understands the underlying cause. A very literal example of an attacker as a scientist who is trying to understand a phenomenon, i.e. the system being attacked, can be seen in [1].

## II. MOTIVATION

Imagine a vulnerability identification process that consists of the following: waiting for an information system to be attacked, then, assuming it survives and one can detect the attack, analyzing the attack, and if the information system is still not compromised, adding this information to one's knowledge base. This technique would be unacceptable to most people, but it is essentially the technique used today. Information assurance, and vulnerability analysis in particular, are hard problems primarily because they involve the application of the scientific method by a defender to determine a means of evaluating and thwarting the scientific method applied by an attacker. This self-reference of scientific methods would seem to imply a non-halting cycle of hypothesis and experimental validation being applied by both offensive and defensive entities, each affecting the operation of the other. Information assurance depends upon the ability to discover the relationships governing this cycle

and then quantifying and measuring the progress made by both an attacker and defender. A metric and framework are required for quantifying information assurance in such an environment of escalating knowledge and innovation. Progress in vulnerability analysis and information assurance research cannot proceed without fundamental metrics. The metrics should identify and quantify fundamental characteristics of information in order to guarantee assurance. A fundamental definition of vulnerability analysis is formulated in this paper based upon attacker and defender as reasoning entities, capable of innovation. Truly innovative implementations of attack and defense lead to the evolution of complexity in an information system. Understanding the evolution of complexity in a system enables a better understanding of where to measure and how to quantify vulnerability and leads towards a calculus of information complexity. The design and implementation of a complexity-based technique is presented as a vulnerability analysis tool for automated measurement of information assurance. The motivation for complexity-based vulnerability analysis comes from the fact that complexity is a fundamental property of information and can be ubiquitously applied to determine vulnerability.

## III. CURRENT NOTIONS OF SECURITY

There have many attempts to define security models that facilitate the proof of security properties [11]. The results that will be presented in this paper focus upon what has been termed probabilistic, rather than possibilistic, security. Possibilistic security is concerned with proofs that given security properties can never be violated, while probabilistic security is concerned with estimating the likelihood that properties will be violated. The quantification of the insecurity that results from the successful exploitation of areas of weak security is referred to in this paper as vulnerability. The security framework generally assumed is that there are low-level and high-level users within a system. The intuitive notion is that high-level users should be secure from low-level users. Security properties include *noninference*: low-level users should not be able to infer information about high-level users, *noninterference*: high-level users are prevented from influencing the behavior of low-level users (otherwise, low-level users could infer information about high-level user activity), *non-deducible output*: low-level users cannot distinguish the events causing high-level users' output, and finally *separability*: no interaction or information flow is allowed between low and high level users. Separability is too strong a security property because it does not allow low-level users to interfere with high-level users. This type of interference is acceptable, since

---

<sup>1</sup> Extended Summary Submitted for review to the DIMACS Workshop on Complexity and Inference, June 2–6 2003. This work is funded by DARPA Fault Tolerant Networks Project contract F30602-01-C-0182.

it is assumed that information flow is allowed from low-level to high-level users. The *perfect security* property allows information to flow only from low to high-level users. While in theory these properties are useful in attempting to prove that a system is secure, anecdotal evidence suggests that few developers will expend the effort to ensure that their systems meet these properties. The number of events that must be verified for possibilistic security results in a combinatorial explosion. Thus probabilistic methods are required; this work attempts to develop a quantification of the degree to which a system has achieved perfect security using fundamental properties of information, rather than proving perfect security. Security properties such as noninference, noninterference, non-deducible output, and separability define various mechanisms by which information flow, that is, information that could be inferred by one class of user about another class of user, is prevented. Similarly results in this work are based upon information flow generated by a low-level user, referred to as an attacker, inferring information about higher-level users. However, it is assumed that security is not discrete, but varies throughout a system and that attackers will tend to follow paths of least resistance. Probabilistic security has been explored in the past; however, obtaining values for probabilities of insecurity has generally been ill defined. This paper will develop a new framework using Kolmogorov Complexity as an underlying means to estimate insecurity probabilities.

#### IV. PROJECT SUMMARY

One goal of our effort has been the development of complexity-based vulnerability analysis (CBVA) utilizing a complexity-based information assurance metric for vulnerability analysis. The metric proposed is based upon Kolmogorov Complexity. Computable estimates of Kolmogorov Complexity have been indicated, as well as additional useful applications of Kolmogorov Complexity for communications in general. Unless vulnerabilities can be identified and measured, the information assurance of a system can never be properly designed or guaranteed. Results from a study on complexity evolving within an information system (Active Network) using Mathematica [10], Swarm, and a new Java complexity probe toolkit [4][5][7] have been obtained. An underlying definition of information security will be hypothesized based upon the attacker and defender as reasoning entities, capable of learning to outwit one another. This leads to a study of the evolution of complexity in an information system and the effects of the environment upon the evolution of complexity. Understanding the evolution of complexity in a system enables a better understanding of how to measure and quantify the vulnerability of a system.

A significant result of this work has been the proposition that complexity plays a critical role in security and can be broadly applied as the basis for security analysis and design. Tools based upon this paradigm do not require detailed *a priori* information about known attacks, but rather compute vulnerability based upon an inherent, underlying property of information itself, namely, it's Kolmogorov-Chaitin complexity. Another significant result is the definition and relationship between complexity theory and brittle systems

theory and their mutual role in analyzing the vulnerability of system design. Future work will be to further refine the complexity-based vulnerability analysis tool and complexity probes including the addition of results from Brittle Systems Theory. The author suggests that active networks [2][3] provide an ideal platform for studying the interaction and properties of algorithmic information and static sequences of data because data can change form deep within, and as it travels through, the network.

#### V. COMPLEXITY-BASED VULNERABILITY ANALYSIS CHALLENGES

Each component of active network contains probe points through which bit level input and output can be collected. Complexity estimates based upon simple inverse compression ratios have been used. The intent has been to experiment with better complexity measures as the research continues. Consider the complexity of bit-level input and output strings concatenated together. That is, observe an input sequence to an arbitrary process (i.e. a potentially vulnerable process) at the bit-level and concatenate with an output sequence at the bit-level. This input/output concatenation can be applied to entire systems or to components of a system. If there is low complexity in the input/output observations, then it is likely to be easy for an attacker to "understand" that component; thus, a naïve vulnerability metric could be defined as the

inverse of  $\frac{\int_{t_s}^{t_e} \hat{K}(x_t) dt}{(t_e - t_s)}$ , where  $x_t$  is a bit-sequence observed at

time  $t$ ,  $t_s$  is the initial observation time, and  $t_e$  is the end time of a potential attacker's observation of the bit sequence. Because of inherently inaccurate complexity metrics, all our figures tended to show a rising complexity with the number of accumulated observations. However, the rate at which the complexity rises varied significantly.

Complexity-based vulnerability analysis faces many challenges. In particular, the length of time required to obtain an accurate sample (and to obtain it in real-time) is critical. In addition, a stream of data on a network link can be sampled at many possible protocol layers. What effect do layers of protocol and encoding have upon the complexity of an information stream? Can these effects be treated as noise and filtered out? The attacker may view protocol and data at a variety of levels: bit-level, system call level, object code level, source code level, script level, etc... All of these levels coexist simultaneously and need to be considered as part of the Kolmogorov Complexity Map of the system. "Map" is probably an incorrect term, because given the simultaneous coexistence of layers of protocol and encoding in an information stream, a three dimensional view (that includes depth through layers of encoding) might be more apt. A potential attacker is looking for areas of complexity low enough to understand and control (yet that provides a path to a potential target) but also, in some cases, high enough in which to potentially hide activity. However, the ability for the attacker to successfully hide his activity depends upon obtaining a good understanding of the system first.

## Index of Terms-- Information Assurance, Kolmogorov Complexity, and Active Networks.

### REFERENCES

- [1] Albert-Laszlo Barabasi, Vincent W. Freeh, Hawoong Jeong & Jay B. Brockman, *Parasitic Computing*, Nature, Vol. 412, 30 August 2001, www.nature.com, pages 894-897.
- [2] Bush, Stephen F. and Kulkarni, Amit B. *Active Networks and Active Virtual Network Management Prediction: A Proactive Management Framework*. ISBN 0-306-46560-4. Kluwer Academic/Plenum Publishers. Spring 2001.
- [3] Bush, Stephen, F. and Kulkarni, Amit B., and Evans, Scott C. *Active Virtual Network Management Prediction Enhancement via Kolmogorov Complexity Estimation*, Submitted as GE CRD Technical Report.
- [4] Evans, Scott, Bush, Stephen F., and Hershey, John. *Information Assurance through Kolmogorov Complexity*. DARPA Information Survivability Conference and Exposition II (DISCEX-II 2001). 12-14 June 2001. Anaheim, California.
- [5] Evans, Scott and Bush, Stephen. F. *Symbol Compression Ratio for String Compression and Estimation of Kolmogorov Complexity*. Submitted to 2002 IEEE International Symposium on Information Theory.
- [6] Kirchner W., Li M., and Vitanyi P., *The Miraculous Universal Distribution*. The Mathematical Intelligencer, Springer-Verlag, New York, Vol. 19, No. 4, 1997.
- [7] Kulkarni, Amit B., Bush, Stephen, F., and Evans, Scott C., *Detecting Distributed Denial-of-Service Attacks using Kolmogorov Complexity Metrics*. Submitted as GE CRD Technical Report.
- [8] Ming Li and Paul Vitanyi. *Introduction to Kolmogorov Complexity and Its Applications*. Springer-Verlag, 1993. ISBN 0-387-94053-7.
- [9] Shannon, C.E. *A mathematical theory of communication*. Bell System Technical Journal, vol. 27, pp. 379-423 and 623-656, July and October 1948.
- [10] Wolfram, Stephen. *Mathematica ...A System for Doing Mathematics by Computer*. Addison-Wesley, Reading, MA, USA, second edition, 1991.
- [11] Zakinthinos, Aris. *On The Composition of Security Properties*. University of Toronto, 1997. Ph.D. Thesis.