Cyber Security – Transition from Science and Art to Engineering.

Brian Witten
Symantec Corporation

Computer security has had a mixed history, beginning with strong formalisms and absolute terms which provided strong security in isolated vaults even before the Internet revolutionized connectivity in a world where "Security was not a requirement of the protocol." Then, as the speed of the Internet's growth surprised even it's creators, countless efforts were made to apply security as an after-thought, with Emergency Response Teams, "firewalls," intrusion detection, anti-virus, anti-spam, anti-spyware, and even anti-phishing, not to mention a list of countless others, added post facto, sans metrics initially, with only the simplest measures of effectiveness coming slowly into focus.

This talk is intended to describe, from an engineer's perspective, how early we "cyber security professionals" are in the process of a transformation from "art" that is neither measurable nor reliably repeatable, and well grounded theories which either cannot scale or do not provide useful specificity, toward useful, scalable, and reliably repeatable disciplines. Some argue that because "security" is organized against intelligent and malicious adversaries, that security can never be truly engineered. Perhaps there will always be a bit of the "Art of War" in cyber security. However, even the warcraft of tanks, planes, and ships are thoroughly engineered with well mapped capabilities, limitations and vulnerabilities. Perhaps cyber security is on similar path.

Aside from the overview, and the setting of context as above, the talk is to be given mainly by a short series of examples, with very few historical examples, but rather a strong focus on some of the newer and newest defenses, along with their metrics and performance, disconcerting thoughts about how some of the metrics are measured, and only a very small discussion of possibilities for how some of these pieces could come together in the future.

Specific examples to be covered include:

Intrusion detection – Early systems could only detect known attacks, and even the means of detection for known attacks could be evaded by adding proper noise. Newer systems are capable of detecting previously unseen attacks with some measures of statistical reliability that are sensitive to factors such as propagation vectors and growth curves, as well as sensor distribution and a number of latencies.

AntiSpam – Billions of legitimate email per sent each day. Yet, the volume of spam outweighs the volume of legitimate mail by a vast margin. Moreover, with such volumes, models of legitimate mail change relatively slowly while the corpus of spam changes so quickly that new characterizations must be distributed several times an hour to continue blocking even just 95% of spam while blocking not more than 00.0001% of legitimate mail. Although the Receiver Operator Characterization of Anti-Spam is not novel, the degree to which temporal aspects of the "move/counter-move" "game" are captured by a simple latency is interesting, particularly as these models seem quite resilient, and increasingly resilient, to evasive "salting." However, as spammers shift from evasion to mimicry, the models begin to increasingly rely on authentication, currently acknowledged as inadequate.

Recovery – It is interesting to note that some interesting similarities between the AntiSpam and Intrusion Detection examples given above. Both are self modifying to detect emergence of new threats within their area of responsibility. Both provide statistical performance in terms of reliability of protection. Moreover, unless traffic is delayed,[1] those statistics are sensitive to the timing of "when" a potential victim is exposed to a threat, or at least the timing of their exposure relative to other potential victims. For these, and many other reasons, recovery technologies are increasingly popular. Most recovery technologies roll back to last known good state. Some recovery technologies also allow "nearly transparent" continued operation through rapid or parallel reprocessing of filtered input. Such filtering often results in measurable dropping of records or connections. However, disconcertingly, given that the "last known good state" included the vulnerability causing the system to be vulnerable to the attack, the system remains vulnerable to such measurable loss induction until some form of self-modifying defense reduces the losses associated with such recovery.

Software Development – Feasible, bug free software would eliminate many (admittedly not all, but many) of the computing industry's security problems. Historically, "correct by construction" has struggled to scale up to the needs of a feature hungry software industry. Also, "bug finding" tools have struggled to find enough of the bugs with reasonable effort to have the scale of effect desired. Recently, newer tools have shown that it's possible to analyze millions of lines of code in a matter of hours, with false positives outnumbered by true positives, and against thoroughly (manually) analyzed target code-bases, true-positives representing a substantial fraction of known ground truth. However, many of the measures here are fraught with pitfalls. Such measures include "defect density," defect distributions, lines of code analyzed per hour, and most disconcertingly for now, the reality that known ground truth is always a subset of ground truth. However, better automating the process of finding and fixing bugs pre-release _should_ reduce the number of harmful bugs found post release, but that hypothesis remains to be validated.

[1] In some cases it is possible to effectively delay traffic. In other cases it is not possible, and in yet other cases delaying the traffic also slows the detection algorithm with no net positive effect.

Of course, there is much work yet to be done. To the best of my knowledge, there is no single, generally accepted, fully normalized, “orthogonal” representation of how these many threats and countermeasures fit together. Yet, such taxonomy has been the long craved “next step” for many years, even as generation after generation of threats appears. However, as threats appear for each protocol, for better or worse, we are finding a larger zoo to canonical-ize each year. However, at least now, we’re beginning to measure our progress against each class of threat in somewhat meaningful ways. True, the measurements are contextually sensitive, as are many engineering measures, and at least newer defenses are decreasingly susceptible evasion, making the measures of protection against each class of threat “somewhat meaningful.”

Now if only adversaries would quit inventing threats! However, given recent emergence of SpIM, which is Spam for Instant Messaging, and SPIT, which is Spam for IP Telephony, perhaps the adversaries will only quit inventing threats after the world quits perpetuating increasingly useful technologies. However, at least now we have far broader experience base of “proven strategies and measures” to bring to bear against each new threat.