Input to the Commission on Enhancing National Cybersecurity

Submission date: September 6, 2016

Joint submission made by:

Benjamin Gittins
Chief Technical Officer
b.gittins@synaptic-labs.com
+356 9944 9390

Ronald Kelson
Chief Executive Officer
r.kelson@synaptic-labs.com
+356 9944 9390

Synaptic Laboratories Ltd.
www.synaptic-labs.com
13 Nadur Heights,
Nadur NDR-1390,
MALTA, Europe

Designers of safe and secure computing and communication architectures. Developers of general-purpose soft IP for FPGA devices, to increase security and performance, and to reduce circuit area.

Topic of this submission:

Design and Deploy Universally Trustworthy and Dependable Systems To Build Trust Between Mutually Suspicious Participants To Minimise Reliance On Deterrence Strategies and Power Struggle

RFI topic areas this submission relates to:

- Cybersecurity Research and Development
- Cybersecurity Insurance
- Critical Infrastructure Cybersecurity
- Identity and Access Management
- Internet of Things
- International Markets

Input submission contents:

(1) A 1 page executive summary for this comment, in the format requested by the RFI, which "identifies the topic addressed, the challenges, and the proposed solution, recommendation, and/or finding." Citations in the Executive Summary map back to the references listed at the end of the 8 page cyber security article attached to this submission. We have inserted headings that match these points in the executive summary.

(2) A supporting 8 page article. R. Kelson and B. Gittins. "Trustworthy Systems that Leverage Distrust Amongst Sovereigns." An Invited Cyber Security Presentation to the WAAS, ELN, and NATO International Conference on Nuclear Threats and Security, 2012. (The URL's have been updated in this version)

(3) Brian Snow. We need assurance! In ACSAC '05: Proceedings of the 21st Annual Computer Security Applications Conference, pages 3–10, Washington, DC, USA, Dec. 2005. IEEE Computer Society. Full text published online on the ACASC website. (https://www.acsac.org/2005/papers/Snow.pdf)

*If "**offence**" capabilities rely on the existence of systems that asymmetrically control exposure to hazards for the profit of one group, then it seems reasonable to assert that "**defence**" capabilities rely on the existence of systems that symmetrically control exposure to hazards in a way that seeks to protect the legitimate interests of all stakeholders.*

— Synaptic Laboratories Limited

Design and Deploy Universally Trustworthy and Dependable Systems To Build Trust Between Mutually Suspicious Participants To Minimise Reliance On Deterrence Strategies

1 Page Executive Summary.

RFI Topics: Cybersecurity Research and Development, Cybersecurity Insurance, Critical Infrastructure Cybersecurity, Identity and Access Management, Internet of Things, International Markets

The Challenges: Modern societies are almost totally dependent upon cyber systems that are evidently not safe or secure. To paraphrase [7] the Director of the U.S. National Security Agency (NSA): there is no such thing as secure anymore... we must assume the attacker is or can get inside our systems (2010). Successful cyber-physical attacks can strike instantly, destroying critical infrastructure, including nuclear power facilities (e.g. Stuxnet virus) [8]. Due to the scale of potential (financial and physical) damage from such cyber attacks, any of these activities could fuel an escalation to nuclear war – particularly if physical destruction from a cyber attacks coincides with a conventional conflict situation [22].

We argue that deterrence in a cyber-security context is *intrinsically* destabilising. For deterrence to be "credible", it is required to introduce a continuous psychological impact on the intended target. To quote [Defense Science Board (DSB). Resilient Military Systems and the Advanced Cyber Threat. U.S. Department of Defence, Jan. 2013.]: "The intellectual and empirical underpinnings for strategy and doctrine for kinetic, nuclear, counterterrorism, counterinsurgency, and other missions have been extensively documented and debated for decades. ... Relatively little has been documented or extensively debated concerning offensive cyber operations. This is especially true with respect to **the use of offensive capability as a component of a larger strategic deterrence that, to be effective, must achieve visible results against the adversary but not reveal enough about the capability for an adversary to create a defense."** … "To prevent the threat of cyber attack … in the global economic and political system, no strategic competitor or adversary can be allowed to gain … offensive cyber superiority. … Current trends, however, could lead some of our country's adversaries to *believe* that their offensive cyber capabilities, together with **their mission-critical defensive postures, are sufficient to neutralize current U.S. conventional or nuclear force capabilities**, Cyber offense is both an enabler for military operations and, as argued in previous chapters, is a critical rung in the escalation ladder for U.S. deterrence strategy."

In short, **when any organisation is invested in employing deterrents, that organisation is also invested in ensuring *all other parties* are vulnerable to attacks.** In short, **the use of cyber deterrence strategies by one or more countries perpetuates cyber vulnerabilities and global instability.**

Clearly, systems involving humans or computers fail in countless ways. We cannot rely on any "entity" to behave consistently in our best interest or even in its own best interest. **Therefore, we are compelled to build trustworthy systems from potentially untrustworthy entities**. One technique is to employ separation of powers [2] as employed in political systems. In principle, if one branch of state malfunctions, the other two (or more) can limit the damage and rebalance the system. In the social theory of power, the system of checks-and-balances involves the participation of all levels of organisation down to the individual.

The solution: Today it is technically feasible to design credible regional, international, and global-scale Information and Communication Technology (ICT) systems that drastically reduce reliance on deterrence strategies and power struggle to build trust between the mutually suspicious participants. These are programmatic systems that do **not** rely on the altruism of any party, and that **simultaneously leverage distrust between active participants to create trustworthy systems**. In particular, these systems combine the security controls employed in democratic political systems in combination with high-assurance fault-tolerant computing techniques. The attached paper provides a worked example – an introduction to International Identity and Access Management solution designed by Synaptic Labs that can win support amongst traditional enemies to deliver a reasonable and higher level of universal trust ***for all stakeholders***.

The recommendation: We respectfully propose that the Commission's detailed recommendations to strengthen cybersecurity should include the following points (that we argue in greater detail in the Peer-Reviewed Technical Publication attached to this submission):

1. Perform an in-depth survey to identify, catalogue and evaluate **the cyber-defence strategies** (which may include the use of defence, deterrence and offence components)**,** employed by the top 10 most powerful nations. Perform a high-level Failure Mode and Effects Analysis with regard to likely outcomes of the individual application of and the interaction between those defence strategies with regard to all levels of stakeholders (e.g from individuals located in any country up to the global system of nation states) in multi-jurisdiction, multi-stakeholder Internet-scale environments. Quantify the costs to the global community of policy outcomes that intrinsically result in global instability. Quantify the benefits to the global community of policy outcomes that promote global stability. Develop a framework that assists policy makers create intrinsically stable cybersecurity policy.

Sincerely, Benjamin Gittins and Ronald Kelson

Originally published in WAAS, ELN, and NATO International Conference on Nuclear Threats and Security, 2012. Page numbers added and corrections to URLs made in this revision (2016 Sep)

Trustworthy Systems that Leverage Distrust Amongst Sovereigns

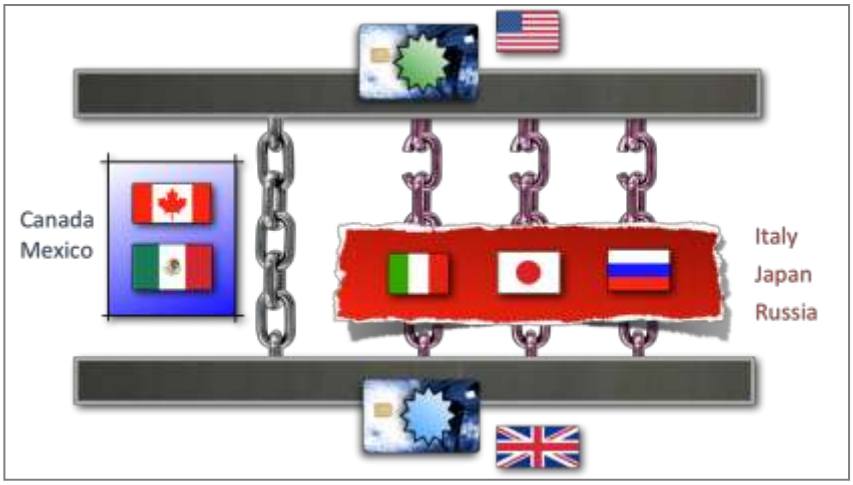
Ronald Kelson
Chairperson & CEO, Synaptic Laboratories Limited, Malta
Vice Chair, ICT Gozo Malta

Benjamin Gittins
Chief Technical Officer, Synaptic Laboratories Limited, Malta
Chief Technical Officer, ICT Gozo Malta

Abstract

Modern societies are almost totally dependent upon cyber systems that are not safe or secure. To paraphrase [7] the Director of the U.S. National Security Agency (NSA): there is no such thing as secure anymore... we must assume the attacker is or can get inside our systems (2010). Successful cyber-physical attacks can strike instantly, destroying critical infrastructure, including nuclear power facilities (e.g. Stuxnet virus) [8]. Many cyber attacks defy accurate attribution [4]. They can gain access to top secret intelligence, industrial control systems, components required to support and/or build nuclear bombs, and so on [20]. Due to the scale of potential (financial and physical) damage from such cyber attacks, any of these activities could fuel an escalation to nuclear war – particularly if physical destruction coincides with a conventional conflict situation [22]. See [23] for global cyber status survey.

As demonstrated by the recent strategy driven nonviolent struggles around the world, grievance groups are increasingly successful at over-throwing powerful institutions that do not uphold the legitimate interests of that grievance group, even in the face of severe sanctions, violent repression, and even death (Dr. Gene Sharp [1]).



We believe it is possible to design credible regional, international, and global-scale Information and Communication Technology (ICT) systems that drastically reduce reliance on deterrence strategies and power struggle to build trust between the mutually suspicious participants. These are pragmatic systems that do not rely on the altruism of any party, and that simultaneously leverage distrust between active participants to create trustworthy systems.

This paper discusses threshold based computer systems that can securely distribute power across sovereign service providing entities, in which each entity: a) provides services ($) to the community, b) can guarantee their own security, and c) gains increased security when they collaborate with other entities that are either strangers, competitors, and in particular hostile adversaries [14]. The political tension that discourages collusion between the service

provider entities (nations) can be exploited to provide higher assurances of security to all clients/stakeholders:

This new ICT architecture adapts political techniques that were originally designed to reduce fear between humans of unequal power.

1. Are there alternatives to (violent) sanctions?

In our view it is possible to design and deploy global-scale systems (that perform formally-defined and agreed services for all stakeholders) that:

a) drastically reduce reliance on deterrence strategies (the threat of sanctions/terror) or on power struggle (the threat of severing the institutions various sources of power) to build trust between the mutually suspicious participants; and

b) do not rely on the altruism of any of the parties, but rather simultaneously leverage any pair-wise association of: integrity, (unilateral or mutual) distrust, and even outright hostility between participants, to create trustworthy systems.

Over-riding self-interest within various institutions and organisations has resulted in the global deployment of, and dependency on, fundamentally insecure computing and communication systems. However, the same myopic self-interests can be intelligently leveraged to begin to make these systems safer in a manner whereby each participating nation state can trust in its own security controls, but gains stronger security through collaboration with other nation states, where each state's security becomes like an independent, redundant strand in a woven steel rope. In this model, any one strand is strong enough. The model can scale up to create international, multi-jurisdiction, global-scale ICT systems. A similar model can also be adapted all the way down to singular computer chips.

2. Tell me again, what has ICT got to do with nuclear deterrence?

Today ICT is as essential as water and electricity. We are all reliant on the same hardware, software, protocols and systems. Unfortunately, today's ICT infrastructure is not trustworthy and cannot be depended upon. To quote [5]: "*[Security] Threats to cyberspace pose one of the most serious economic and* ***national security challenges*** *of the 21st Century for the United States and our allies*."

To quote Melissa Hathaway (who led [5]): "In director *[ed. of U.S. National Intelligence] Blair's testimony to the Senate in February, he stated: 'The national security of the U.S., [and] our economic prosperity [is] threatened.'* ***And I would say that it is compromised.****"* (2010) [6] To quote Debora Plunkett, Director of the Information Assurance Directorate (IAD), U.S. NSA: "There is no such thing as **Secure** anymore." (2010) [7], [8].

To quote Isaac Ben-Israel, Director of the Defense R&D Directorate in the Israel Ministry of Defense (1998-), *"If you want to hit a country severely you hit its power and water supplies*. ***Cyber technology can do this without shooting a single bullet.****" (2012)* [9].

Unlike nuclear weapons, generally speaking cyber-attacks against critical infrastructure can originate from any location, and successfully strike within the Observation-Orientation-Decision-Action loop [7] of human defenders.

Latent vulnerabilities and malware, sometimes deliberately built in at point of manufacture, could be exploited at any time. Fundamental vulnerabilities in the conceptual design of these systems are well known inside expert circles.

To quote Brian Snow, former Technical Director of the U.S. NSA IAD for 12 years: *"The creators of the Internet knew that MALICE was a serious issue." ... "However, [they]*

pushed security aside due to the perceived difficulties, or cost, and that is the start of our problems today. To put it bluntly, the Internet was not built to address the known risks. By design, the Internet naïvely relies on the honesty of every network user, and places far too little emphasis on healthy mutual suspicion! The cost and risks were not eliminated -- rather they were both shifted away from the designers and the manufacturers, and transferred to the Global user base. You and me pick up the check!" (2012) [17]

To quote a security expert from CISCO on the Civilian Identity Management Infrastructure: *"In practice is it snake oil? It is somewhat indistinguishable [ed. from placebo] in practice because of the problems."* (2010) [14]

To quote a Director at the U.S. Center for Strategic & International Studies (CSIS) [8]:

"The electrical grid. A popular target in the military." ... "If I was a hacker, and I hacked into the control system, kinda like stuxnet, of one of these big huge room-sized generators, what could I do to it?

The answer is: you can make it jump up and down, emit smoke, and shake itself to pieces."

To quote a Former U.S. NSA Director's Fellow [21]:

- An attack could bring down the electricity grid for 6 months;
- This would lead to no communications, no banking, and food production ceasing.
- It would require months to bring the country back online.

See [23] for a high level survey of expert opinions wrt. the known problems undermining today's ICT ecosystem.

3. Cyber attacks on critical infrastructure as a catalyst leading to nuclear war

According to the World Economic Forum's Global Risks 2012 Report [19]: Critical systems failure was identified as *"a key concern for world leaders from government, business and civil society" and that this will "most likely be caused by cyber attacks"*. Today, cyber attacks rank 4th out of 50 global risks.

Today, potentially more than 140 countries have a cyber weapon development programme. Many nation states, acting out of fear, will imitate DARPA's global cyber-offensive "Plan X" [10], [11].

It is exceeding difficult to assign attribution to cyber attacks [4]. A cyber attack may *appear* to originate from a specific computer. However that computer may have been compromised with malware, malware that is under third party control and forwarding the attack without the legitimate owner's knowledge or consent. Attacks can be relayed through many computers and countries.

We only have to imagine a modern "Cuban missile crisis" like situation in which an anonymous third party starts destroying critical infrastructure in either the USSR or the U.S. The situation will aggressively escalate if at first viewing it appears that the attack originates from the other country. However, it is quite possible that this other country's computers are *also* compromised, and simply form a link in a chain where the ultimate attacker is beyond identification or reach during the critical go/no go decision window of any retaliation or preemptive strike by either of those two countries.

We need an effective global-scale inclusive common cyber defence that does not rely on the threat or use of violent sanctions. Key objectives are to design ICT systems:

1. from the onset take into account the human trust factor to manage known risks;
2. that maintain integrity when latent faults or undetected malware are exploited; and
3. that employ parallelism and redundancy where each instance is independent (sovereign) and sufficiently secure; in which some non-trivial number of different instances must be broken to break the system.

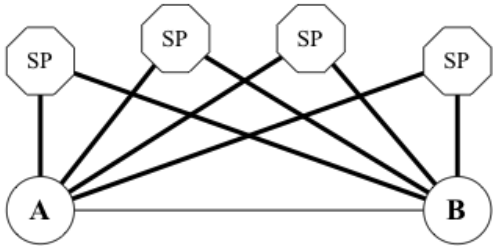
Synaptic Labs has successfully pursued advanced research and design based on these approaches.

4. Decentralising power

Systems, involving humans or computers fail in countless ways. We cannot rely on any "entity" to behave consistently in our best interest. Dictatorships are a prime example of "single point of trust failure". We must build trustworthy systems from potentially untrustworthy entities. One technique is to employ separation of powers [2]. In principle, if one branch of state malfunctions, the other two (or more) can limit the damage and rebalance the system – this being the key goal.

5. Decentralising trust in computer systems

In 1976, Hellman, Diffie and Lamport proposed a simple computer system [12] that decentralised power. In this case, instead of relying on 1 service provider (SP), the burden of responsibility was distributed over 4 (or more) service providers. Unlike in "human systems", in computing it is trivially easy for many computers to perform **exactly** the same operation.



This computer system was designed to provide "identity management" and "secure communication" services. User A could ask to send a message to User B. User B would receive that message and receive 4 assertions regarding the identity of the sender. Without going into the technical details of *how*, for the purpose of *privacy* only 1 service provider had to behave honestly with regard to users A and B. For the purpose of *availability* you could deploy the system to remain operational in the face of 1, 2 or even 3 *simultaneously exploited* arbitrary faults (collusion or third party attack).

6. Leveraging distrust to increase trustworthiness

The goal is to create a decentralised system of nodes that avoids imploding on itself (resulting in a centralised system) or exploding (disintegrating). When power is decentralised across entities, we want to ensure each entity wishes to participate in the system but not collude, and ensure that the system is tolerant to arbitrary operational faults.

All systems that decentralise power are a type of *threshold* system. After some threshold is met or exceeded, it is assumed the correct decision has been made (e.g. taking the consensus opinion regarding a question decided by vote).

The system's integrity is compromised if some number of entities greater than or equal to that threshold, are coerced into colluding together as a single entity in a malicious way.

A problem with popular democratic systems is that individual stakeholders typically cannot ensure their security acting unilaterally. This can expose minority groups to prejudices of the majority group. This occurs when democratic principles are misapplied as a tool to decide "*what is in the best interest of the majority*" as opposed to deciding *"what is in the best interest of all stakeholders*."

What is fascinating in Hellman, Diffie and Lamport's 1976 proposal, is that security (privacy) can be maintained by the presence of just ONE honest service provider, even when all other ($N - 1$) participating service providers are colluding. Modern invasion of privacy is a silent/covert failure: we do not know when it is happening, and so we must seek the greatest assurances that it is not happening. In contrast, a divergent decision by one or two parties is a visible/overt failure. This visibility of failure on each client transaction notifies the stakeholder(s) in question and permits them to make a choice to substitute a new service provider for the "faulty" service provider (this can be automated). We make 2 observations regarding tension between service providers:

1. Every entity participating as a service provider can ensure it's own privacy;
2. If a service provider X is a large organisation, the participation of the ($N - 1$) other service providers offers security against insider attacks performed by X's staff provisioning that service.

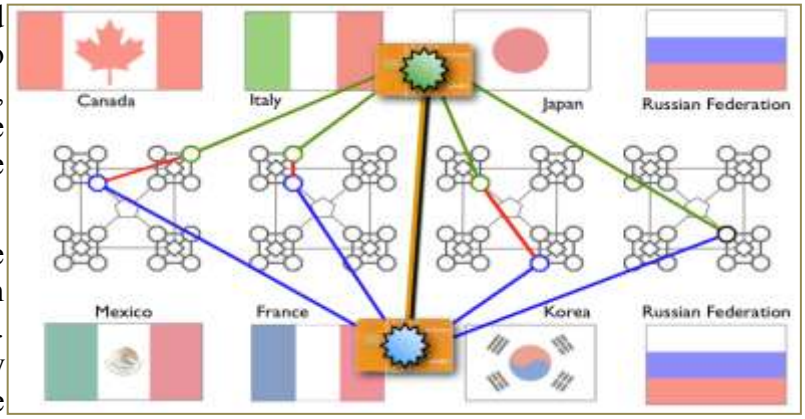
Additional properties can be achieved:

1. A service provider X gains increased assurances that other service providers will not collude against X by ensuring competitors and opponents of those service providers are actively participating in X's client transactions.
2. In global-scale systems, most stakeholders (clients) using the system are NOT service providers. Those stakeholders gain increased assurances the system will protect their legitimate interests if the service providers are strangers to each other, fierce opponents, or preferably adversaries.
3. If each service provider is also a client of the system, they have the ultimate reason not to collude. This increases security assurances for all stakeholders of the system.
4. An unassociated attacker must breach the security of at least 4 independently secure service providers before they can breach the security of the end users' transaction (or attack the clients computer directly).
5. Synaptic Labs' TruSIP computer is designed to to provide similar types of security fault tolerance to the client's and service provider's computers [23]. (Protect all stakeholders.)

7. A scalable decentralised ICT System: A simplified one-page description

In this section we offer a simplified description of part of our peer-reviewed [13] global scale identity management service cited in [18] at NATO. Find a highly accessible video presentation of this technology online at the 2010 IEEE Key Management Summit [14]. Also see [15], [16]. *This system has greater flexibility, security, and capabilities, than briefly described here*.

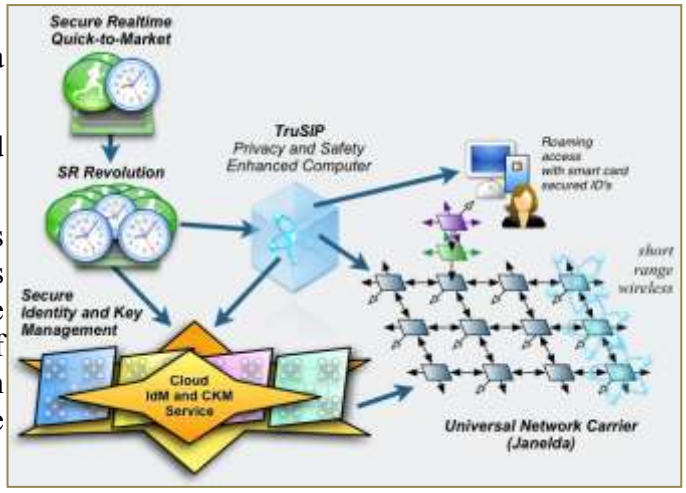
Just like in the Hellman, Diffie and Lamport 1976 proposal, we also **distribute trust** over *N* different entities, in this case we replace "service providers" with confederations of service providers:



Hundreds of different service providers, from different countries, can be members within each "confederation". Advantageously, client transactions only need to employ *at most* 2 service providers from each confederation to enable secure services between them. If one service provider is compromised, or goes rogue, only a small subset of the stakeholders are potentially effected.

In this hypothetical configuration, the system **maintains privacy** for the end users so long as none of the service providers in one confederation colludes with one service provider from each of the other confederations. In short, collusion is difficult due to existing political tensions.

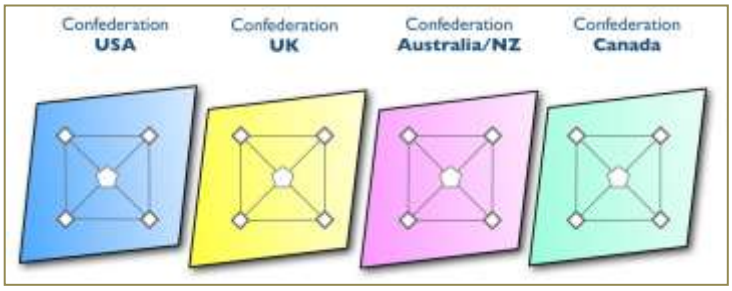
This simplified system is strengthened in a variety of additional ways not described here.



8. Foundations of a trustworthy and dependable ICT ecosystem

In the same way that some Governments employ a wide variety of political techniques (originally [2]) with the goal of protecting the legitimate interests of its citizens, the creation of a trustworthy and dependable ICT ecosystem requires a variety of different techniques to be adapted to the particulars of each component.

Synaptic Labs has been working ~12 years studying the open hard safety and security problems in today's ICT ecosystem, and designing commercially viable solutions in fields ranging from safe and secure realtime computing (used for critical infrastructure applications) through to a next-generation network to improve the security and performance of today's Internet system.



Our technologies are explicitly product and vendor neutral. In many cases our solutions can be adopted by today's market leading ICT companies to harden the next generation of *their* existing product families.

Synaptic Labs' goal is to protect as much of today's existing ICT software and hardware as possible at the least cost. Visit http://ictgozomalta.eu and [23] to learn more.

9. Closing statement

Todays's ICT ecosystem was not built to be trustworthy or dependable [6]. Cyber-physical attacks against critical infrastructure can lead to situations that escalate to nuclear war. As countries become increasingly cognisant of their almost total dependence on today's ICT ecosystem [5], countries will seek to protect their sovereignty and "secure their interests". Fear has already driven many countries to develop cyber-offensive [10], [11] capabilities as a deterrence strategy. It is difficult to attribute the true origin of cyber attacks, making accountability difficult, sanctions complicated, and opportunities for abuse high.

What is required is an inclusive global-scale ICT ecosystem that encourages mutually suspicious entities to collaborate in a way that results in a system that seeks to protect the legitimate interests of all stakeholders, irrespective of their relative power relationship, without reliance on violent sanctions. We have shown how to adapt the spirit of some political techniques in the architecture of a global-scale Identity Management ecosystem. It is the authors experience, that almost any ICT system can be hardened to be much more trustworthy and dependable.

Any entity supporting the design, development and deployment of these approaches will increase regional, national and global stability by improving the trustworthiness of our common ICT foundations, and by building a more stable base from which to reduce our perceived dependency on, and desire to own, nuclear weapons.

References

1. G. Sharp. *There Are Realistic Alternatives*. The Albert Einstein Institution, December 2003. https://librivox.org/there-are-realistic-alternatives-by-gene-sharp/
2. B. d. M. de Secondat, Charles. The Spirit of the Laws.
3. F. Osinga. Science, Strategy and War, The Strategic Theory of John Boyd. Universiteit Leiden, Jan. 2005.
4. UK Strategic Defence and Security Review. 2010.
5. Cyberspace policy review, United States, May. 2009.
6. M. Hathaway. Plenary speaker. CSIIRW-6 ORNL, 2010.
7. J. Wolf. U.S. code-cracking agency works as if compromised. Newspaper article, Reuters, Dec. 2010.
8. AtlanticLIVE. The atlantic and government executive cyber security forum. Video, The Atlantic, 2010.
9. Grauman. Cyber-security: The vexed question of global rules. Security & Defence Agenda, Brussels, Jan. 2012.
10. Fed Biz Opps.Gov, DARPA-SN-12-51, 2012.
11. Ellen Nakashima, With Plan X, Pentagon seeks to spread U.S. military might to cyberspace, The Washington Post, 2012.
12. W. Diffie and M. E. Hellman. Multiuser cryptographic techniques. In AFIPS '76, June 1976. ACM.
13. B. Gittins. Overview of SLL's proposal in response to NIST's call for new global IdM/CKM designs without public keys. In Proceedings of the CSIIRW-6, ORNL, 2010. ACM.
14. B. Gittins and R. Kelson. Overview of SLL's proposal in response to NIST's call for new global IdM/CKM designs without PKC. Video. In IEEE Key Management Summit 2010 website.
    https://www.youtube.com/watch?v=8Z3Prkc2eng

15. B. Gittins. Outline of a proposal responding to E.U. and U.S. calls for trustworthy global-scale IdM and CKM designs. Report 2011/029, Cryptology ePrint Archive, 2011.
16. B. Gittins and R. Kelson. Feedback to NIST DRAFT Special Publication 800-130. Comment, August 2010.
17. B. Snow, Our Security Status is Grim (and the way ahead will be hard), Video. Nov. 2011.
18. O. McCusker, et al. Combining Trust and Behavioral Analysis to Detect Security Threats in Open Environments. In NATO IACDS 2010, RTO-MP-IST-091, April 2010.
19. Global Risks 2012, Insight Report. World Economic Forum, seventh edition, 2012.
20. J. Brenner. America the Vulnerable - Inside the new threat matrix of digital espionage, crime and warfare. Penguin Press HC, The, Sep. 2011.
21. O. S. Saydjari. Addressing the Nation's Cyber Security Challenges: Reducing Vulnerabilities Requires Strategic Investment and Immediate Action. Testimony before the House Committee on Homeland Security Subcommittee, Apr. 2007.
22. Beyond War - A New Way of Thinking, Handbook, 1985. Editors: Dr Martin Hellman, et al.
23. Synaptic Labs 2012 Annual Cyber Status video & slideshow series:
    These 2012 videos and slideshows are currently not available online.
    Please contact Synaptic Labs on info@synaptic-labs.com to express your interested in receiving access to this information.

We Need Assurance!

Brian Snow
U. S. National Security Agency
bdsnow@nsa.gov

Abstract

When will we be secure? Nobody knows for sure – but it cannot happen before commercial security products and services possess not only enough functionality to satisfy customers' stated needs, but also sufficient assurance of quality, reliability, safety, and appropriateness for use. Such assurances are lacking in most of today's commercial security products and services. I discuss paths to better assurance in Operating Systems, Applications, and Hardware through better development environments, requirements definition, systems engineering, quality certification, and legal/regulatory constraints. I also give some examples.

1. Introduction

This is an expanded version of the "Distinguished Practitioner" address at ACSAC 2005 and therefore is less formal than most of the papers in the proceedings.

I am very grateful that ACSAC chose me as a distinguished practitioner, and I am eager to talk with you about what makes products and services secure.

Most of your previous distinguished practitioners have been from the open community; I am from a closed community, the U.S. National Security Agency, but I work with and admire many of the distinguished practitioners from prior conferences.

I spent my first 20 years in NSA doing research developing cryptographic components and secure systems. Cryptographic systems serving the U.S. government and military spanning a range from nuclear command and control to tactical radios for the battlefield to network security devices use my algorithms.

For the last 14 years, I have been a Technical Director at NSA (similar to a chief scientist or senior technical fellow in industry) serving as Technical Director for three of NSA's major mission components: the Research Directorate, the Information Assurance Directorate, and currently the Directorate for Education and Training (NSA's Corporate University). Throughout these years, my mantra has been, "Managers are responsible for doing things right; Technical Directors are responsible for finding the right things to do."

There are many things to which NSA pays attention in developing secure products for our National Security Customers to which developers of commercial security offerings also need to pay attention, and that is what I want to discuss with you today.

2. Setting the context

The RSA Conference of 1999 opened with a choir singing a song whose message is still valid today: "Still Haven't Found What I'm Looking For". The reprise phrase was . . . *"When will I be secure? Nobody knows for sure. But I still haven't found what I'm looking for!"*

That sense of general malaise still lingers in the security industry; why is that? Security products and services should stop malice in the environment from damaging their users. Nevertheless, too often they fail in this task. I think it is for two major reasons.

First, too many of these products are still designed and developed using methodologies assuming random failure as the model of the deployment environment rather than assuming malice. There is a world of difference!

Second, users often fail to characterize the nature of the threat they need to counter. Are they subject only to a generic threat of an opponent seeking some weak system to beat on, not necessarily theirs, or are they subject to a targeted attack, where the opponent wants something specific of theirs and is willing to focus his resources on getting it?

The following two simple examples might clarify this.

Example 1: As a generic threat, consider a burglar roaming the neighborhood wanting to steal a VCR. First, understand his algorithm: Find empty house

(dark, no lights) try door; if open, enter, if VCR – take. If the door is resistant, or no VCR is present, find another dark house.

Will the burglar succeed? Yes, he will probably get a VCR in the neighborhood. Will he get yours? What does it take to stop him? Leave your lights on when you go out (9 cents a kilowatt-hour) and lock your door. That is probably good enough to stop the typical generic burglar.

Example 2: As a targeted threat, assume you have a painting by Picasso worth $250,000 hanging above your fireplace, and an Art thief knows you have it and he wants it. What is his algorithm? He watches your house until he sees the whole family leave. He does not care if the lights are on or not. He approaches the house and tries the door; if open, he enters. If locked, he kicks it in. If the door resists, he goes to a window. If no electronic tape, he breaks the glass and enters. If electronic tape is present, he goes to the siding on the house, rips some off, then tears out the fiberboard backing, removes the fiberglass insulation, breaks though the interior gypsum board, steps between the studs, and finally takes the painting and leaves.

It takes more effort to counter a targeted threat. In this case, typically a burglar alarm system with active polling and interior motion sensors as a minimum (brick construction would not hurt either). With luck, this should be enough to deter him. If not, at least there should be increased odds of recovery due to hot pursuit once the alarms go off.

There is no such thing as perfect security; you need to know how much is enough to counter the threat you face, and this changes over time.

3. What do we need?

NSA has a proud tradition during the past 53 years of providing cryptographic hardware, embedded systems, and other security products to our customers. Up to a few years ago, we were a sole-source provider. In recent years, there has come to be a commercial security industry that is attractive to our customers, and we are in an unaccustomed position of having to "compete." There is nothing wrong with that. *If* industry can meet our customer's needs, so be it.

Policy and regulation still require many of our customers to accept Government advice on security products. However, they really press us to recommend commercial solutions for cost savings and other reasons. Where we can, we do so. However, we do not do it very often because we still have not found what *we* are looking for – assurance.

Assurance is essential to security products, but it is missing in most commercial offerings today. The major shortfall is absence of assurance (or safety) mechanisms in *software*. If my car crashed as often as my computer does, I would be dead by now.

In fact, compare the software industry to the automobile industry at two points in its history, the 1930s and today. In 1930, the auto industry produced cars that could go 60 mph or faster, looked nice, and would get you from here to there. Cars "performed" well, but did not have many "safety features." If you were in an accident at high-speed, you would likely die.

The car industry today provides air bags, seat belts, crush zones, traction control, anti-skid braking, and a host of other safety details (many required by legislation) largely invisible to the purchaser. Do you *regularly* use your seat belt? If so, you realize that users *can* be trained to want and to use assurance technology!

The software security industry today is at about the same stage as the auto industry was in 1930; it provides performance, but offers little safety. For both cars and software, the issue is really assurance.

Yet what we need in security products for high-grade systems in DoD is more akin to a military tank than to a modern car! Because the environment in which our products must survive and function (battlefields, etc.) has malice galore.

I am looking forward to, and need, convergence of government and commercial security products in two areas: assurance, and common standards. Common standards will come naturally, but assurance will be harder – so I am here today as an evangelist for assurance techniques.

Many vendors tell me that users are not willing to pay for assurance in commercial security products; I would remind you that Toyota and Honda penetrated U.S. Markets in the 70's by differentiating themselves from other brands by improving reliability and quality! What software vendor today will become the "Toyota" of this industry by selling robust software?

4. Assurance: first definition

What do I mean by assurance? I'll give a more precise definition later, but for now it suffices to say that assurance work makes a user (or accreditor) more confident that the system works as intended, without flaws or surprises, even in the presence of malice.

We analyze the system at design time for potential problems that we then correct. We test prototype devices to see how well they perform under stress or when used in ways beyond the normal specification. Security acceptance testing not only exercises the product for its expected behavior given the expected

environment and input sequences, but also tests the product with swings in the environment outside the specified bounds and with improper inputs that do not match the interface specification. We also test with proper inputs, but in an improper sequence. We anticipate malicious behavior and design to counter it, and then test the countermeasures for effectiveness. We expect the product to behave safely, even if not properly, under any of these stresses. If it does not, we redesign it.

I want functions *and* assurances in a security device. We do not "beta-test" on the customer; if my product fails, someone might die.

Functions are typically visible to the user and commanded through an interface. Assurances tend to be invisible to the user but keep him safe anyway.

Examples would be thicker insulation on a power wire to reduce the risk of shock, and failure analysis to show that no single transistor failure will result in a security compromise.

Having seat belts in a car provides a safety function. Having them made of nylon instead of cotton is the result of assurance studies that show nylon lasts longer and retains its strength better in the harsh environment of a car's interior.

Assurance is best addressed during the initial design and engineering of security systems – not as after-market patches. The earlier you include a security architect or maven in your design process, the greater is the likelihood of a successful and robust design. The usual quip is, "He who gets to the interface first, wins".

When asked to predict the state of "security ten years from now," I focus on the likely absence of assurance, rather than the existence of new and wonderful things.

Ten years from now, there will still be security-enhanced software applications vulnerable to buffer overflow problems. These products will not be secure, but will be sold as such.

Ten years from now, there will still be security-enhanced operating systems that will crash when applications misbehave. They will not be secure either.

Ten years from now, we will have sufficient functionality, plenty of performance, but not enough assurance.

Otherwise, predicting ten years out is simply too hard in this industry, so I will limit myself to about five years. Throughout the coming five-year span, I see little improvement in assurance, hence little true security offered by the industry.

5. The current state of play

Am I depressed about this state of affairs? Yes, I am. The scene I see is products and services sufficiently robust to counter many (but not all) of the "hacker" attacks we hear so much about today, but not adequate against the more serious but real attacks mounted by economic enemies, organized crime, nation states, and yes, terrorists.

We will be in a truly dangerous stance: we will think we are secure (and act accordingly) when in fact we are not secure.

The serious enemy knows how to hide his activities. What is the difference between a hacker and a more serious threat such as organized crime? The hacker wants a *score*, and bragging rights for what he has obviously defaced or entered. Organized crime wants a *source*, is willing to work long, hard, and quietly to get in, and once in, wants to stay invisible and continue over time to extract what it needs from your system.

Clearly, we need confidence in security products; I hope we do not need a major bank-failure or other disaster as a wake-up call before we act.

The low-level hackers and "script-kiddies" who are breaking systems today and are either bragging about it or are dumb enough to be caught, are providing some of the best advertising we could ask for to justify the need for assurance in security products.

They demonstrate that assurance techniques (*barely*) adequate for a benign environment simply will not hold up in a malicious environment, so we *must* design to defeat malice. Believe me – there is malice out there, beyond what the "script-kiddies" can mount.

However, I do fear for the day when the easy threats are countered – that we may then stop at that level, rather than press on to counter the serious and pernicious threats that can stay hidden.

During the next several years, we need major pushes and advances in three areas: Scalability, Interoperability, and Assurance. I believe that market pressures will provide the first two, but not the last one – assurance.

There may or may not be major breakthroughs in new security functions; but we really do not need many new functions or primitives – if they come, that is nice. If they do not, we can make do with what we have.

What we really need but are not likely to get is greater levels of assurance. That is sad, because despite the real need for additional research in assurance technology, the real crime is that we fail to

use fully that which we already have in hand! We need to better use those confidence-improving techniques that we do have, and continue research and development efforts to refine them and find others.

I am not asking for the development of new science; the safety and reliability communities (and others) know how to do this – go and learn from them.

You are developers and marketers of security products, and I am sorry that even as your friend I must say, "Shame on you. You should build them better!" It is a core quality-of-implementation issue. The fact that teen-age hackers can penetrate many of your devices from home is an abysmal statement about the security-robustness of the products.

6. Assurance: second definition

It is time for a more precise definition. Assurances are confidence-building activities demonstrating that

1. $ The system's security policy is internally consistent and reflects the requirements of the organization,
2. $ There are sufficient security functions to support the security policy,
3. $ The system functions meet a desired set of properties and *only* those properties,
4. $ The functions are implemented correctly, and
5. $ The assurances *hold up* through the manufacturing, delivery, and life cycle of the system.

We provide assurance through structured design processes, documentation, and testing, with greater assurance provided by more processes, documentation, and testing.

I grant that this leads to increased cost and delayed time-to-market – a severe one-two punch in *today's* marketplace; but your customers are growing resistive and are beginning to expect, and to demand, better products *tomorrow*. They are near the point of chanting, "I'm mad as hell, and I'm not going to take it anymore!"

Several examples of assurance techniques come to mind; I will briefly discuss some in each of the following six areas: operating systems, software modules, hardware features, systems engineering, third party testing, and legal constraints.

7. Operating systems

Even if operating systems are not truly secure, they can at least remain benign (not actively malicious) if they would simply enforce a digital signature check on every critical module prior to each execution. Years ago, NSA's research organization wrote test code for a UNIX system that did exactly that. The performance degraded about three percent. This is something that is doable!

Operating Systems should be self-protective and enforce (at a minimum) separation, least-privilege, process-isolation, and type-enforcement.

They should be aware of and enforce security policies! Policies drive requirements. Recall that Robert Morris, a prior chief scientist for the National Computer Security Center, once said: "Systems built without requirements cannot fail; they merely offer surprises – usually unpleasant!"

Given today's common hardware and software architectural paradigms, operating systems security is a major primitive for secure systems – you will not succeed without it. This area is so important that it needs all the emphasis it can get. It is the current "black hole" of security.

The problem is innately difficult because from the beginning (ENIAC, 1944), due to the high cost of components, computers were built to share resources (memory, processors, buses, etc.). If you look for a one-word synopsis of computer design philosophy, it was and is SHARING. In the security realm, the one word synopsis is SEPARATION: keeping the bad guys away from the good guys' stuff!

So today, making a computer secure requires imposing a "separation paradigm" on top of an architecture built to share. That is tough! Even when partially successful, the residual problem is going to be covert channels. We really need to focus on making a secure computer, not on making a computer secure – the point of view changes your beginning assumptions and requirements!

8. Software modules

Software modules should be well documented, written in certified development environments, (ISO 9000, SEI-CMM level five, Watts Humphrey's Team Software Process and Personal Software Process (TSP/PSP), etc.), and *fully* stress-tested at their interfaces for boundary-condition behavior, invalid inputs, and proper commands in improper sequences.

In addition to the usual quality control concerns, *bounds checking* and *input scrubbing* require special attention. For bounds checking, verify that inputs are of the expected type: if numeric, in the expected range; if character strings, the length does not exceed the internal buffer size. For input scrubbing, implement reasonableness tests: if an input should be a single word of text, a character string containing multiple words is wrong, even if it fits in the buffer.

A strong quality control regime with aggressive bounds checking and input scrubbing will knock out the vast majority of today's security flaws.

We also need good configuration control processes and design modularity.

A good security design process requires review teams as well as design teams, and no designer should serve on the review team. They cannot be critical enough of their own work. Also in this world of multi-national firms with employees from around the world, it may make sense to take the national affinity of employees into account, and not populate design and review teams for a given product with employees of the SAME nationality or affinity. Half in jest I would say that if you have Israelis on the design team put Palestinians on the review team; or if Germans are on one, put French on the other. . . .

Use formal methods or other techniques to assure modules meet their specifications exactly, with no extraneous or unexpected behaviors – especially embedded malicious behavior.

Formal methods have improved dramatically over the years, and have demonstrated their ability to reduce errors, save time, and even save dollars! This is an under-exploited and very promising area deserving more attention.

I cite two examples of formal methods successes: The Microsoft SLAM static driver verifier effort coming on line in 2005, and Catherine Meadows' NRL Protocol Analyzer detecting flaws in the IKE (Internet Key Exchange) protocol in 1999. You may have your own recent favorites.

As our systems become more and more complex, the need for, and value of, formal methods will become more and more apparent.

9. Hardware features

Consider the use of smartcards, smart badges, or other hardware tokens for especially critical functions. Although more costly than software, when properly implemented the assurance gain is great. The form-factor is not as important as the existence of an isolated processor and address space for assured operations – an "Island of Security," if you will. Such devices can communicate with each other through secure protocols and provide a web of security connecting secure nodes located across a sea of insecurity in the global net.

I find it depressing that the hardware industry has provided hardware security functionality (from the Trusted Platform Group and others) now installed in processors and motherboards that is not yet accessed or used by the controlling software, whether an OS or an application.

10. Security systems engineering

How do we get high assurance in commercial gear?

a) How can we trust, or

b) If we cannot trust, how can we safely use, security gear of unknown quality?

Note the difference in the two characterizations above: *how we phrase the question may be important*. For my money, I think we need more focus on how to use safely security gear of unknown quality (or of uncertain provenance).

I do not have a complete answer on how to handle components of unknown quality, but my thoughts lean toward systems engineering approaches somewhat akin to what the banking industry does in their systems. No single component, module, or person knows enough about the overall transaction processing system to be able to mount a successful attack at any one given access point. To be successful the enemy must have access at multiple points and a great deal of system architecture data.

Partition the system into modules with "blinded interfaces" and limited authority where the data at any one interface are insufficient to develop a complete attack. Further, design cooperating modules to be "mutually suspicious," auditing and alarming each other's improper behavior to the extent possible.

For example: if you are computing interest to post to accounts there is no need to send the complete account record to a subroutine to adjust the account balance. Just send the current balance and interest rate, and on return store the result in the account record. Now the interest calculating subroutine *cannot* see the data on the account owner, and therefore cannot target specific accounts for theft or other malicious action. We need to trust the master exec routine, but minimize the number of subroutines we need to trust. Yes, I know this is over-simplified, but you get my drift.

In addition, to guard against "unintended extra functionality" within given hardware modules or software routines, the development philosophy needs to enforce something akin to "no-lone zones" in that no single designer or coder can present a "black-box" (or proprietary?) effort to the system design team that is tested only at its interfaces and is then accepted.

Review all schematics and code (in detail, line by line) for quality and "responsive to stated requirement" goals. This review should be by parties independent of the designer. This is expensive, but not

far from processes required today in many quality software development environments to address reliability and safety concerns.

This of course requires all tools (compilers, CAD support, etc.) used in the development environment to be free of malice; that can be a major hurdle and a difficult assurance task in and of itself (remember the Thompson compiler in "Reflections on Trusting Trust, CACM 1983)!

The "Open Source" movement may also provide value in this area. There are pluses and minuses with open source, but from the security viewpoint, I believe it is primarily a plu**s.**

Further architectural constraints may be imposed to make up for deficiencies in certain modules. Rather than (or in addition to) encryption in application processes prior to transmission to other sites which could be bypassed or countered by a malicious operating system, you might require site-to-site transmissions to go through an encrypting modem or other in-line, non-bypassable link encryptors.

Link encryption in addition to application layer encryption is an example of a "Defense in Depth" strategy that attempts to combine several weak or possibly flawed mechanisms in a fashion robust enough to provide protection at least somewhat stronger than the strongest component present.

Synergy, where the strength of the whole is greater than the sum of the strength of the parts, is highly desirable but not likely. We must avoid at all costs the all-too-common result where the system strength is less than the strength offered by the strongest component, and in some worst cases less than the weakest component present. Security is so very fragile under composition; in fact, secure composition of components is a major research area today.

Good *system* security design today is an art, not a science. Nevertheless, there are good practitioners out there that can do it. For instance, some of your prior distinguished practitioners fit the bill.

This area of "safe use of inadequate components" is one of our hardest problems, but an area where I expect some of the greatest payoffs in the future and where I invite you to spend effort.

11. Third party testing

NIST (and NSA) provide third-party testing in the National Information Assurance Partnership Laboratories (NIAP labs), but Government certification programs will only be successful if users see the need for something other than vendor claims of adequacy or what I call "proof by emphatic assertion – Buy me, I'm Good."

If not via NIST or other government mechanism, then the industry must provide *third-party* mediation for vendor security claims via consortia or other mechanisms to provide *independent* verification of vendor claims *in a way understandable by users*.

12. Market/legal/regulatory constraints

Market pressures are changing, and may now help drive more robust security functionality. The emergence of e-commerce in the past decade as a driver for secure internet financial transactions is certainly helpful, as is the entertainment industry's focus on digital rights management. These industries certainly want security laid on correctly and robustly!

I hope citizens will be able to use the emerging mechanisms to protect personal data in their homes, as well as industry using the mechanisms to protect industry's fiscal and intellectual property rights. It is simply a matter of getting the security architecture right.

I wonder if any of the industry consortia working on security for digital rights management and/or electronic fiscal transactions have citizen advocates sitting on their working groups.

Lawsuits might help lead to legal "fitness-for-use" criteria for software products – much as other industries face today. This could be a big boon to assurance – liability for something other than the quality of the media on which a product is delivered!

Recall that failure to deliver expected functionality can be viewed, in legal parlance, as providing an "attractive nuisance" and is often legally actionable.

One example is a back yard swimming pool with no fence around it. If a neighbor's child drowns in it, you can be in deep trouble for providing an attractive nuisance. Likewise, if you do a less than adequate job of shoveling snow from your walk in winter (providing the appearance of usability) you can be liable if someone slips on the ice you left on the surface. Many software security products today are attractive nuisances!

All you need do is to Google "Software Quality Lawsuits" or a similar phrase, and you can find plenty of current examples of redress sought under law for lack of quality in critical software. Do not attempt to manage defects in software used in life-critical applications. Remove them during the development and testing processes! People have died due to poor software in medical devices, and the courts are now engaged; the punitive awards can be significant.

One example of a lawsuit already settled: *General Motors Corp. v. Johnston* (1992). A truck stalled and was involved in an accident because of a defect in a PROM, leading to the death of a seven-year old child. An award of $7.5 million in punitive damages against GM followed, in part due to GM knowing of the fault, but doing nothing.

There are social processes outside the courts that can also drive vendors toward compliance with quality standards.

One of the most promising recent occurrences in the insurance industry was stated in the report of Rueschlikon 2005 (a conference serving the insurance industry). Many participants felt that, "The insurance industry's mechanisms of premiums, deductibles, and eligibility for coverage can incent best practices and create a market for security . . . This falls in line with the historic role played by the insurance industry to create incentives for good practices, from healthcare to auto safety . . . Moreover, the adherence to a set of best practices suggest that if they were not followed, firms could be held liable for negligence."

Bluntly, if your security product lacks sufficient robustness in the presence of malice, your customers will have to pay more in insurance costs to mitigate their risks.

How the insurance industry will measure best practices and measure compliance are still to be worked out, but I believe *differential* pricing of business disaster recovery insurance based in part on quality/assurance (especially of security components) is a great stride forward in bringing market pressure to bear in this area!

13. Summary

In closing, I reiterate that what we need most in the future is more assurance rather than more functions or features. The malicious environment in which security systems must function *absolutely requires* the use of strong assurance techniques.

Remember: most attacks today result from failures of assurance, not failures of function.

Rather than offer predictions, try for a self-fulfilling prophecy – each of us should leave this conference with a stronger commitment to using available assurance technology in products! It is not adequate to *have* the techniques; we must *use* them!

We have our work cut out for us; let's go do it.

In closing, I would like to thank Steven Greenwald, Brad Martin, and Greg Shipley for their insights and help in preparing this article.