Statistics, Nature, and the Future of Information Security
[Dan Geer, ISTS, 11 November 08]

Thank you for the introduction; it is both a pleasure to be here and a responsibility to provide something of substance.

As I said in the abstract, security is a means, not an end, and is thus always changing. In fact, the dominating characteristic of digital security is that very thing, the rate of change. I would go so far as to say digital security is quite possibly the most intellectually challenging profession on the planet. In no other field do the facts mutate at the rate they do in our field, and in no other field are the researchers who produce those changes so very often the enemy. Competition is no more good or bad than is the existence of the neutron—it is simply a fundamental reality -- but I don't know of any "place" where competition is as strong as it is in the security world, a world where there is a new reality every morning before breakfast.

That is not to say that competition is not fierce in many, many places and many, many fields. There is fierce competition everywhere -- from two scavengers looking at the same cadaver to two wildcatters drilling for the same strata to two venture capitalists bidding on the same idea—fierce competition where, in fact, there will be a winner and there will be a loser. I don't see, or see why, we shouldn't find the competition over the digital sphere to be as fierce as it is elsewhere.

The fact is that digital security competition is already as fierce, and it will be fiercer still in the future. There is no important sovereign nation that does not have serious work going on for both digital defense and digital offense. While nation-state level digital offense is clearly the pinnacle of competition, I am going to ignore it this hour. If I knew everything, then I would not be able to tell you anyway. Given that I do not know everything, I'll just stay out of it. I

will, however, say that in the digital offense of this kind, it is the unknown unknowns that will kill you. This is precisely what Sec. of Defense Donald Rumsfeld meant when he said that it is the unknown unknowns that interest him. His comment was, without doubt, the smartest thing a US Cabinet Secretary has said in my lifetime. Every person who made fun of Rumsfeld's view proved, if nothing else, that they were innumerate. At the nation state or at the desktop, the unknown unknowns are why we are here. The unknown unknowns are the hidden predictors of kinds of failure. The urge to stamp them out is understandable, but God help us if we take a bargain that promises no more unknown unknowns.

So, leaving nation state offensive capabilities out of this, let me say what makes a good digital security practitioner: It is a love of knowing how things work and by satisfying that love through knowing how they fail. We, all of us, hunt the unknown unknowns. Our closest kin are the zoologists who both know that cataloging all life on the planet is an epochal challenge that is asymptotically impossible but who, nevertheless, delight beyond all else when a species never before seen is discovered, even if the use of the word "discovered" is as strictly incorrect as saying that it was Columbus who discovered America. Only three years ago an entire family of mammals was "discovered" in a Laotian food market, and many of you will be familiar with William Gibson's iconic remark, "The future is already here; it's just unevenly distributed."

This comes down to the idea of risk, what risk we know about and what risk about which we do not know that we do not know. Amongst the risks we know, it is simply a duty to mount a proportional countermeasure. About the risks which are unknown unknowns, it is a different story.

There is some argument in the digital security community over whether the concept of risk, as defined by other fields, has bearing here or whether, the very rate of change and the hidden quality of that change makes formalizing risk something that we should ignore in favor of paying what attention we can to discipline. My colleague

Brian Chess says, and I quote,

> The notion of risk is attractive to the security community because our discipline offers few absolute guarantees, but a mathematical view of risk often fails to serve security practitioners well. It is a poor foundation for building a case for software assurance. Instead we need to promote security as part of a broader engineering discipline.

In no way do I wish to undercut his development of this thesis. I should, however, acknowledge a bias. Everyone who is a leader in the digital security arena today came to that field from some other field for the simple reason that there was no training when we began. This is rapidly changing; all of us who came in early will be soon replaced by the formally trained. As that happens, the renaissance quality of digital security practice will fade, to be replaced by the excellence that comes with crisp specialization. This is not bad, but it is different. Digital security's core knowledge base has reached the size where new recruits can no longer hope to be competent generalists, and, thus, serial specialization is the only broad option available to them.

I said all that to tell you that, unsurprisingly, I, too, am a retread. My formal training was as a biostatistician with a public health slant. Here is a bit of statistician's advice: All data has bias. The question is whether you can correct for it. My first bias is that numbers do matter, that we cannot hope to make sense of the digital security field without them. I understand Chess' point and that of others who will show that some things can't be measured, who quote Mark Twain ("Figures don't lie, but liars figure"), or who pull out the shibboleth that you can prove anything with statistics. To a degree, people like me have to concur with that latter idea, but the best statistician I ever met, Fred Mosteller, put it this way: "It *is* easy to lie with statistics, but it is easier to lie without them." I've written extensively on this and won't repeat it here.

My other bias is a bias held in common with lawyers and preachers, namely that the path to understanding the world is found in analogies. Just as the practice of law is said to be the search for analogies, so is the practice of the

security professional. For me, digital security has its greatest wealth of fruitful analogies in Nature, and not just in borrowing terms like virus and worm from our friends, the biologists. The public health training I took reinforces me in this view, as does my view that evolution and a Creator are not contradictory. The idea of discarding any behavior or design which does not still contribute to survival in favor of a behavior or design which now does, seems as core in the digital world as it is amongst carbon-based life forms.

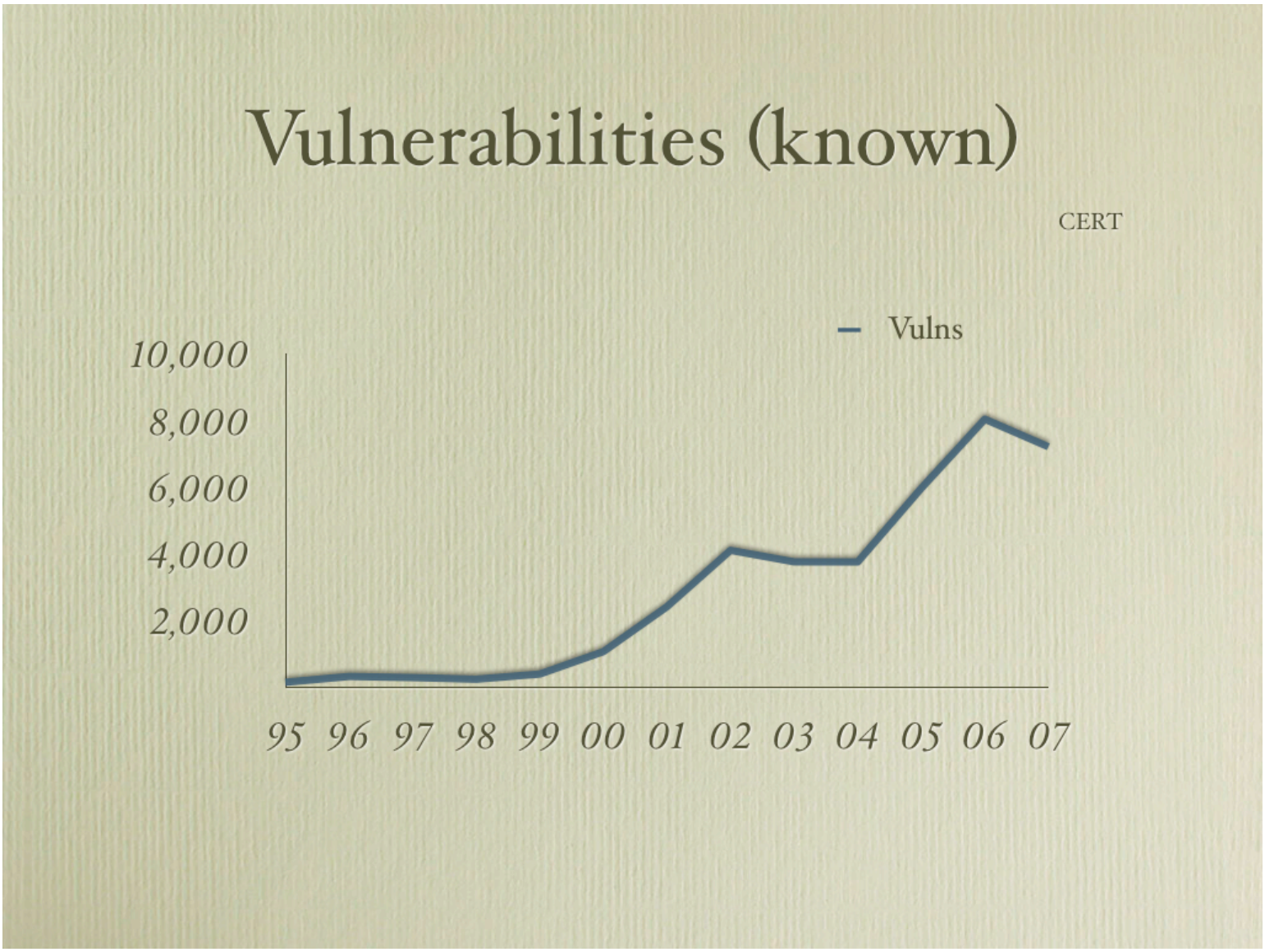
I am somewhat known for championing the idea that digital security is risk management. Risk management, to me, was summed up by Dan Borge in his wonderful _The Book of Risk_. In it Borge makes two points that are fundamental today. One is definitional: that the purpose of risk management is to change the future, not to explain the past. The other is operational: that risk management means taking deliberate action to shift the odds in your favor—increasing the odds of good outcomes and reducing the odds of bad outcomes.

So let me come to what the abstract of this talk promised. We have two main ways to look at the rate of change in digital security and use what we see to predict the, which is to say our, future for the purpose of changing that future: trend analysis of what we know how to measure, and our own re-application of how Nature adapts to what works and what does not work.

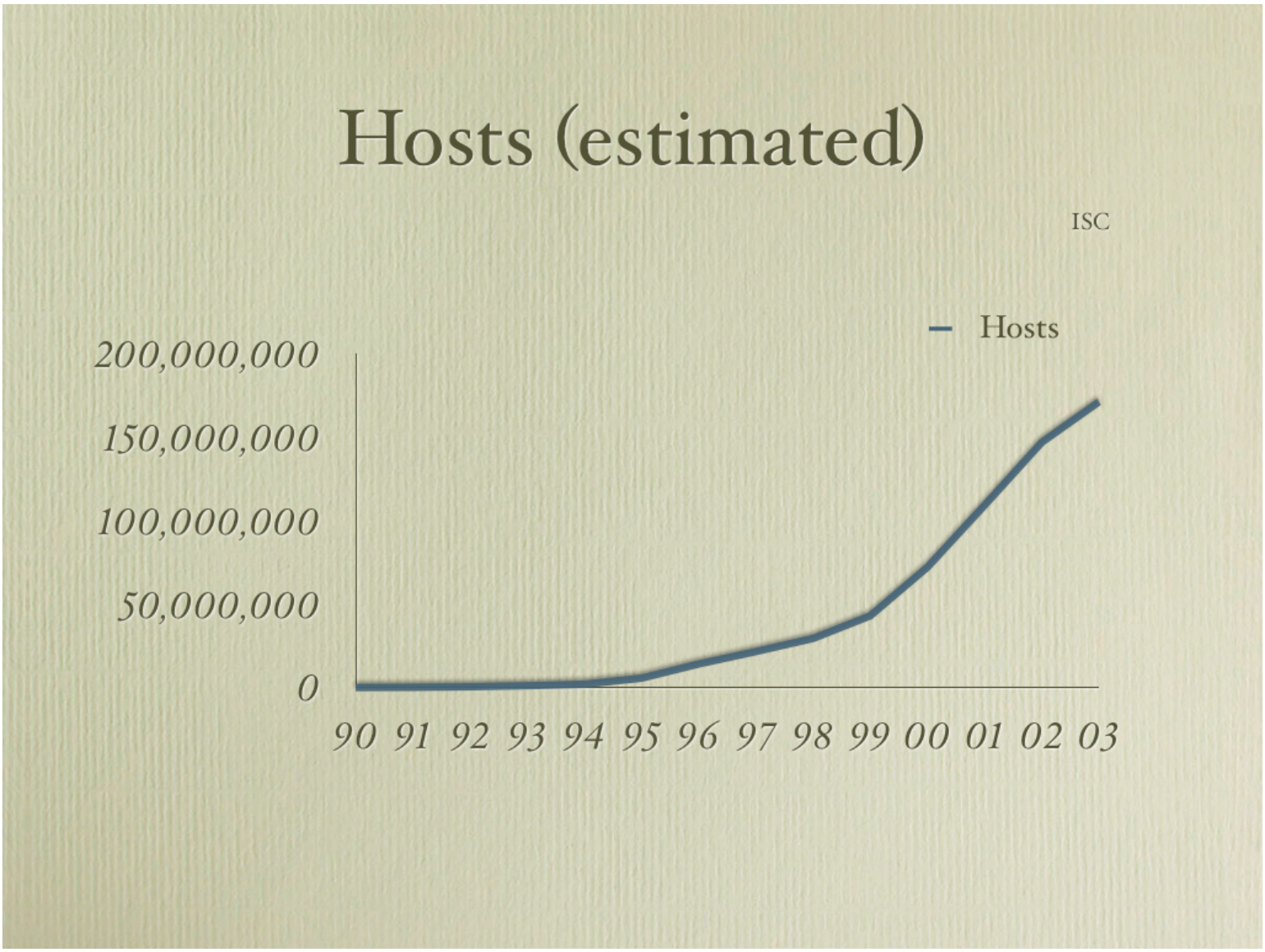
I say "trend analysis" as another minor bit of statistical advice. If we do the kind of security metrics that are available to us at this early stage, they are likely to be trend metrics since, as a point of practice, a poor quality but stable measurement will still give useful trend data absent pathologic details that, for the purpose of this talk, I will ignore. If you, say, regularly undercount the number of vulnerabilities in code by 50%, then you do not have an accurate measure of how vulnerable the code is, but you will have a meaningful measure of whether your code is getting better or worse. This is exactly the problem that, say, a street cop faces in determining if drug enforcement is working or not. That cop almost surely will never know how much heroin is for sale, but

he can measure the price. If the price is falling, then police work may be failing. If it is rising, then police work may be succeeding. The trend, which is to say, the shape of the curve predicts the future even if what is actually being measured has an unknown rate of error. This leaves, however, a problem of interpretation.

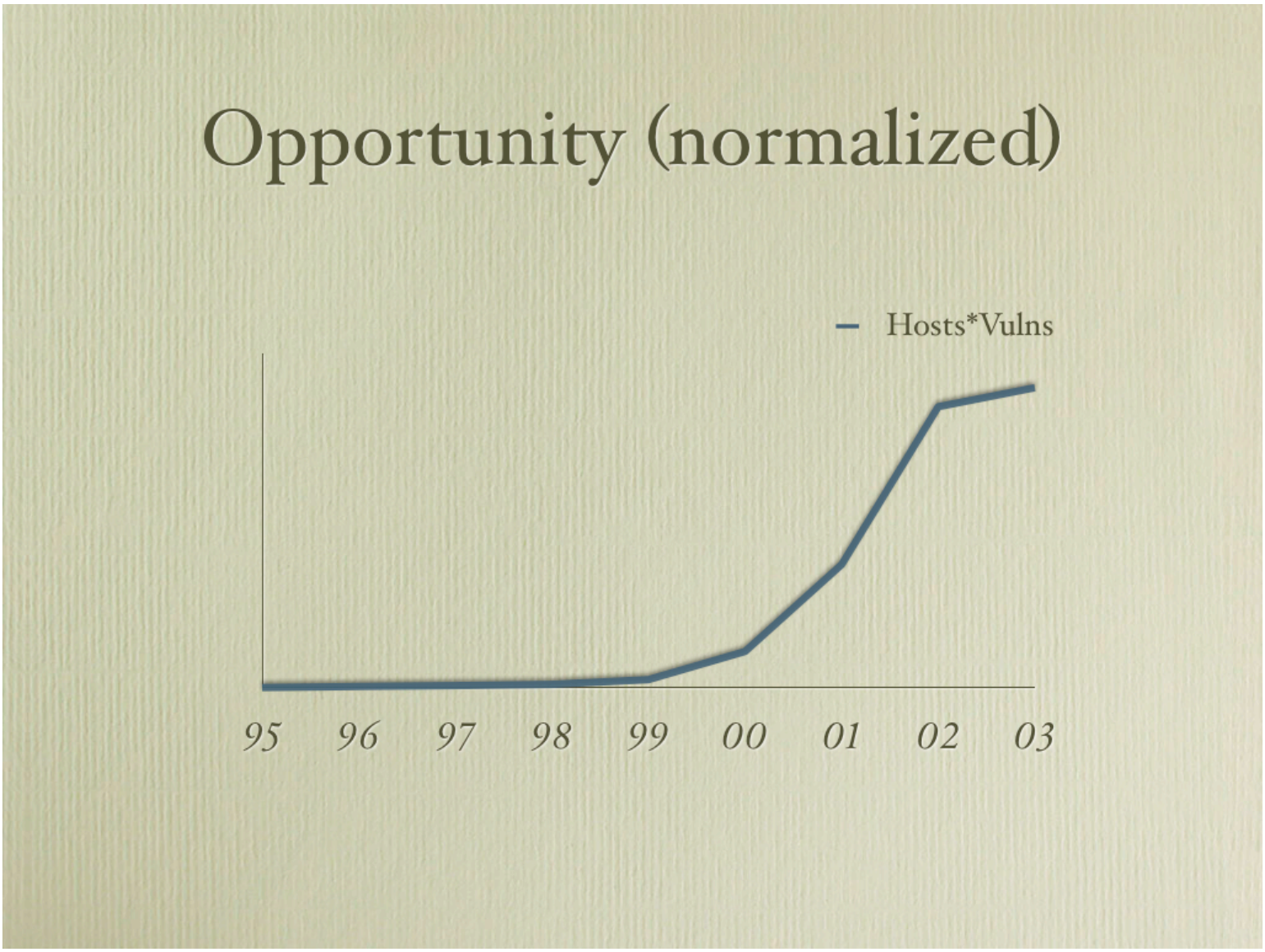
The following slides, ironically based on CERT data, show the incident and vulnerability reporting rates.



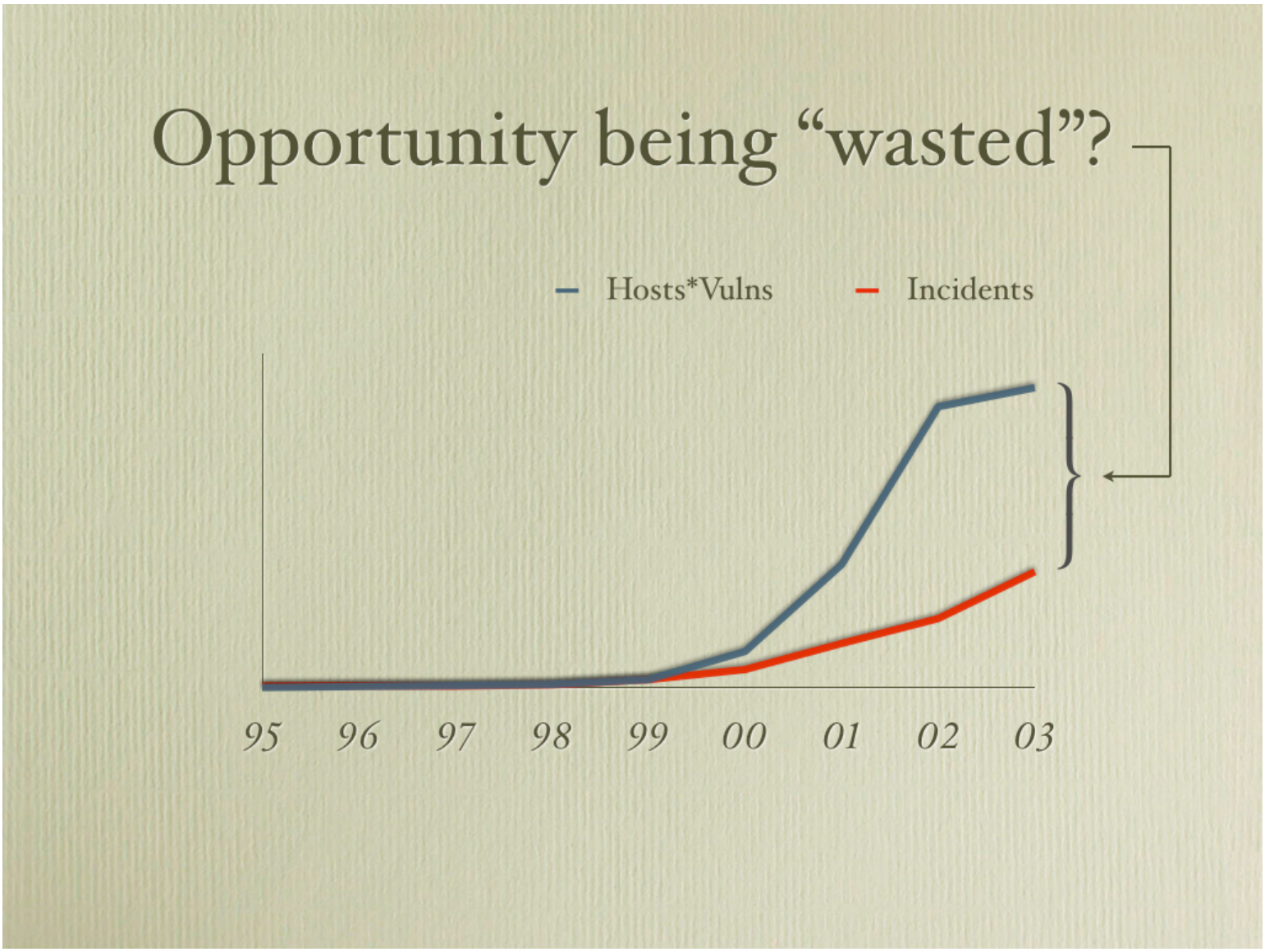
We also have the ISC's estimates of the number of Internet hosts, for the interval up to 2003.



This allows us to ask if the total opportunity space for malware is not proportional to the number of hosts times the number of vulnerabilities.



If that opportunity curve (not arguing about the numbers, just the shape) is close enough to be plausible, then let's normalize both the opportunity curve and the incident curve so that we can plot them on the same access. If we do that, it looks like this.



and the question is then whether the slower rise in incidents versus opportunity a marker for the good guys winning at controlling the rate of growth of e-crime, or is it that the e-crime predators are so satiated that they are leaving prey on the field? Did we push the red-line down, or did the other side just not pull it up? This is an example of the kind of question you can pose with trendlines even when they may be non-believable as absolute measures.

Trend data, however, have a weak point: they will never predict a game changing event, what in evolutionary biology is known as a punctuated equilibrium. That term, due to Stephen Jay Gould, is a reminder that

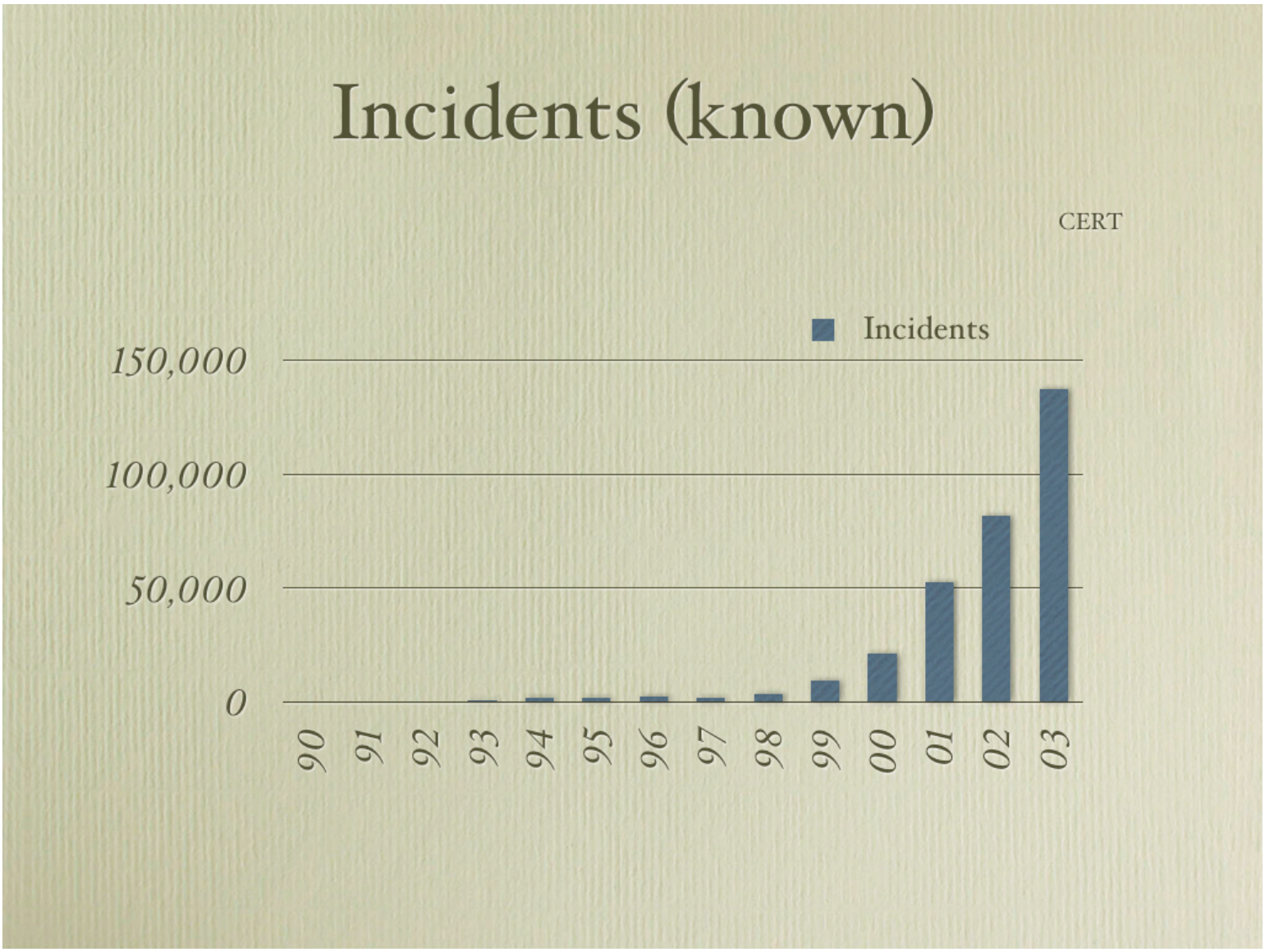
evolutionary change does not come at some smooth rate but is rather characterized by long periods of stability and short periods of rapid change. Such events do not come with much warning, and their echoes may reverberate for a long time as one change begets another. This is not a concept just within biology, as it is approximately the same thing that Nicholas Taleb called a "black swan" event in his book of the same name.

These unexpected events occur in Nature. They also occur in the digital world. They may be random, like a mutation, or they may be an intentional, evil action by an opponent, such as the introduction of a pathogen into an environment previously free of it, whether that environment is a data center or a feedlot. In the short run, it doesn't matter which it is, though in the long run it may well. In fact, if you count unintended consequences, a small change may turn out to be a carrier for a black swan event.

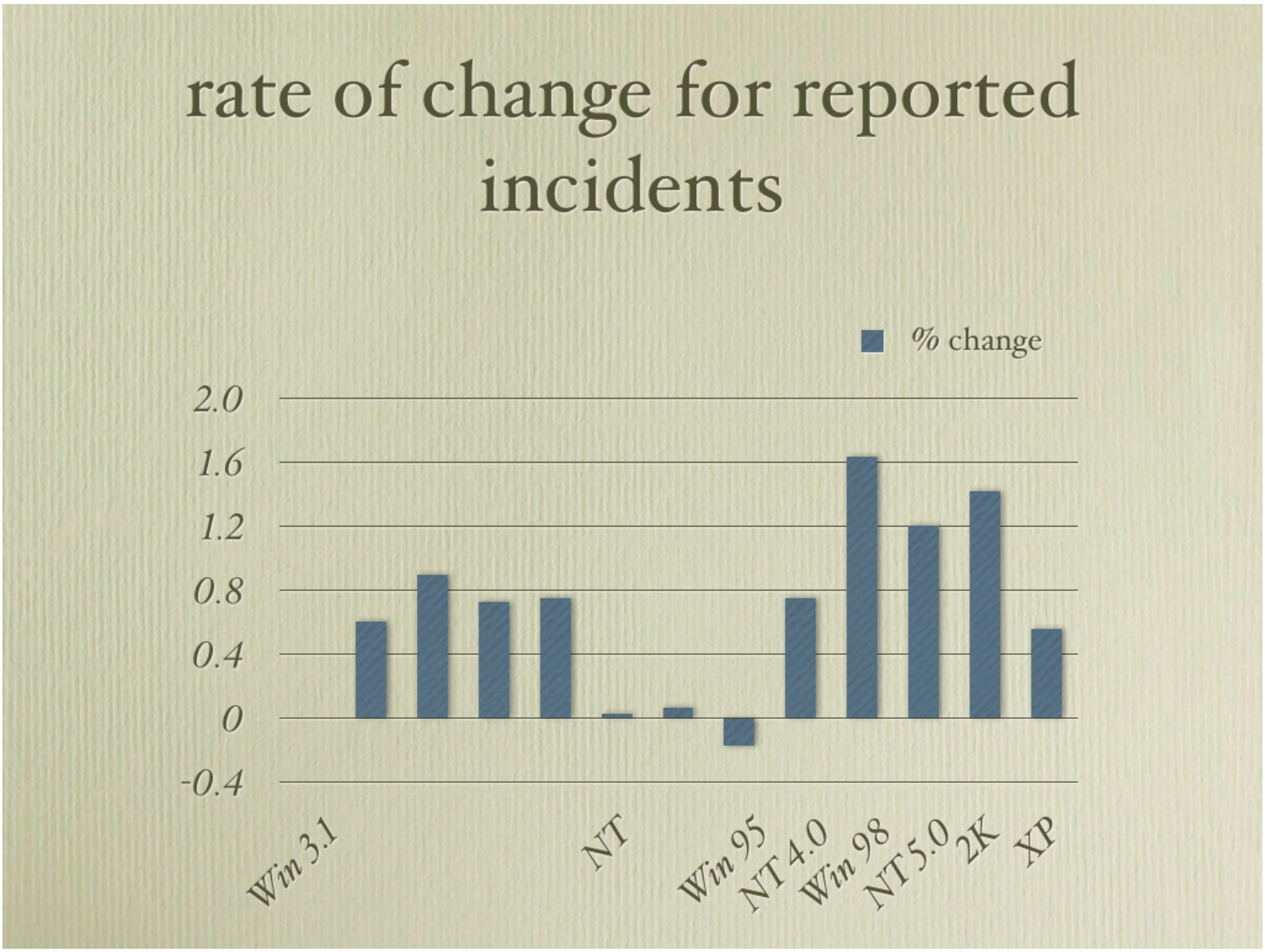
I trace the security industry as we know it today to one of these black swan events, the unintended consequence of a modest, desirable change, a change in comparative slow motion compared to, say, the Witty worm. Specifically, I trace the birth of the industry in which most of you earn your keep to Microsoft's introduction of a TCP/IP stack as a freebie in the Windows platform. That TCP/IP stack took an operating system designed for a single owner/operator on a private net, and connected that OS to the world.

It was well intentioned and inevitable, but it had a side effect that qualifies it as a black swan. Once that stack was installed, every sociopath became a next door neighbor and, as such, we can point to that event as the birth of our industry.

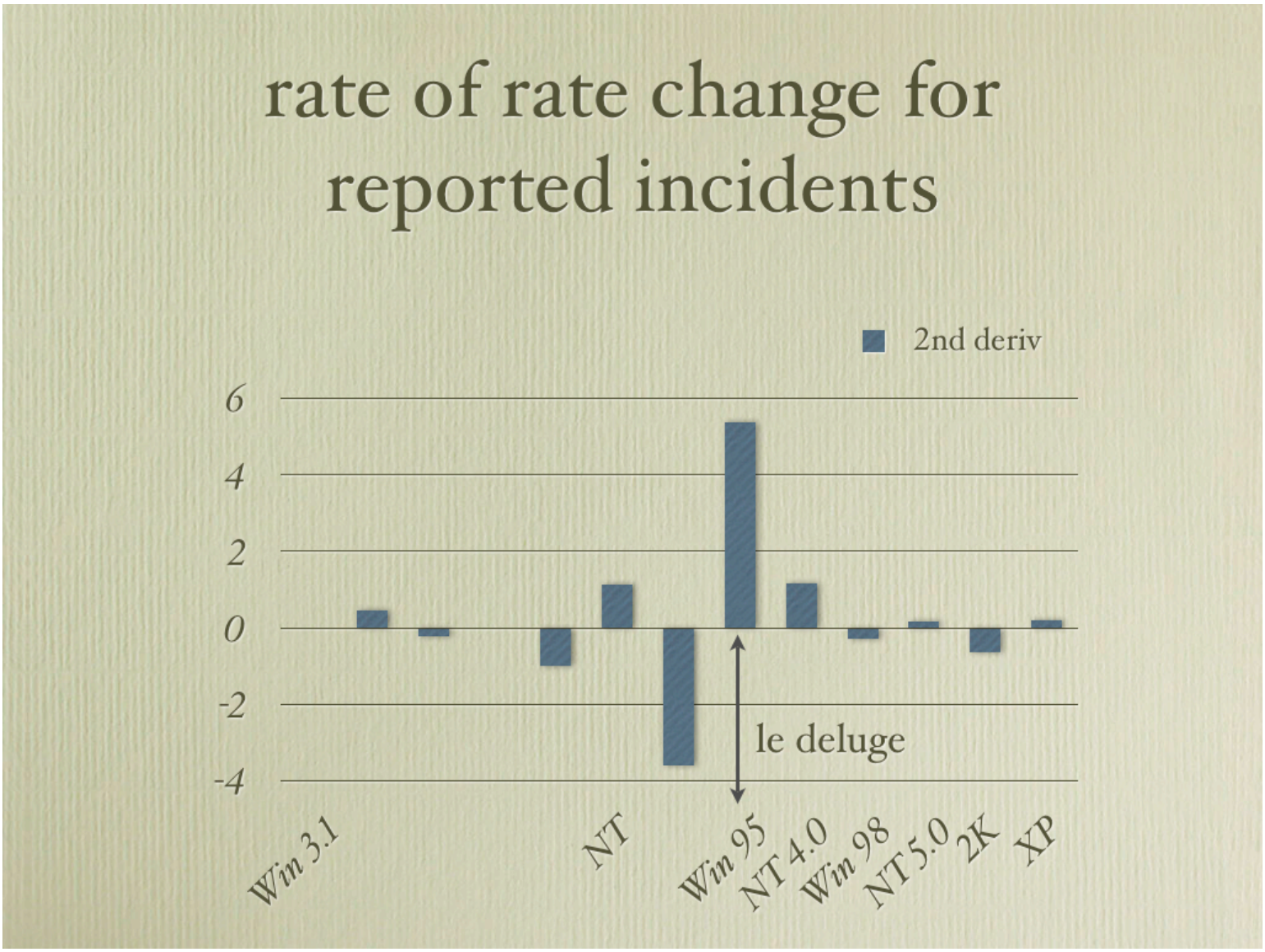
Another picture may be helpful. Again using CERT incident data to illustrate, we redraw that data as a bar graph; it looks like this:



If that incident data is converted from counts to rate of change (the first derivative, if you will) of incident counts, then it looks like the following. Note that we've changed the timeline from calendar year to Microsoft version time.



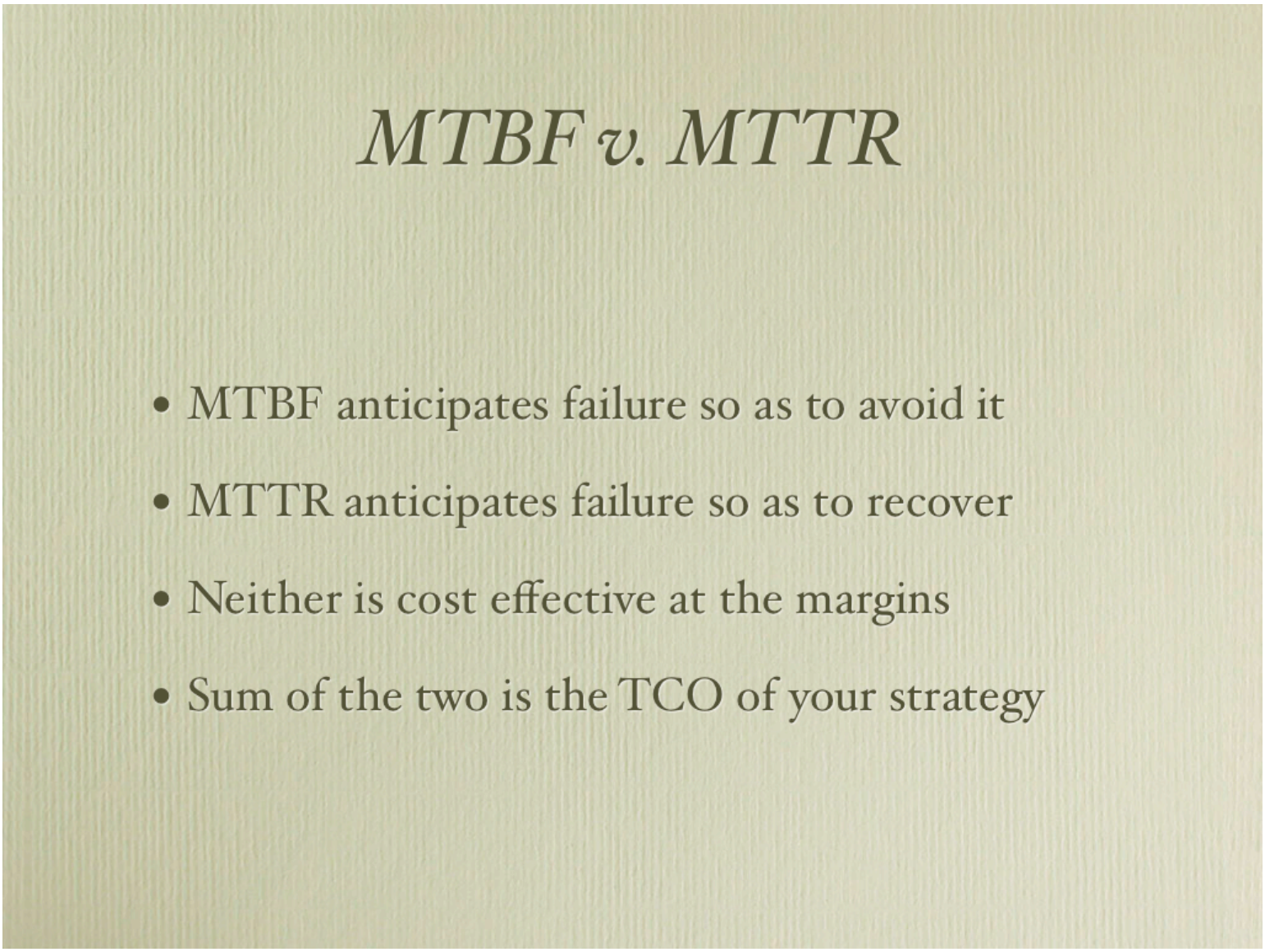
Interesting, but let's go one step further and look at the rate of change of the rate of change (the second derivative, if you will). It looks like this.



Note that the real event, the real equilibrium punctuator, was the introduction of a TCP/IP stack into the installed Windows base. The spike in the second derivative of the rate of attacks underscores the game-changing nature of that event, but nothing much else happened for a while. This was almost like a spontaneous mutation in a genome. The mutation that produced the SARS virus or the species jump that moved AIDS into the human population had no instantaneously visible effect either—that came later. Remember that viable mutations do not go away nor are genies are put back in bottles.

This has impact on risk management in that your risk management strategy has to deal with what the trend analysis says you are most likely to see, but it also needs an out for when the really unexpected occurs.

I suggest that, in mature industries, security is a subset of reliability, which is to say that my definition of security work is the pursuit of "no surprises." Put differently, it is the ability to predict and thus to rely. One way you might put this is to say that your computer system has to be highly available. As you probably know, the calculus of availability is begins with two measuiires, mean time between failures (MTBF) and mean time to repair (MTTR).



As it says, neither MTTR or MTBF is cost effective at the margins, although Availability is perfect at those margins:

Cost Effectiveness & Availability

US Army

$$A = 1 = \begin{cases} \frac{\text{MTBF}}{0+\text{MTBF}} & \text{when MTTR} = 0 \\ \frac{\infty}{\text{MTTR}+\infty} & \text{when MTBF} = \infty \end{cases}$$

In other words, which is more CE to approach, zero recovery time or infinite uptime?

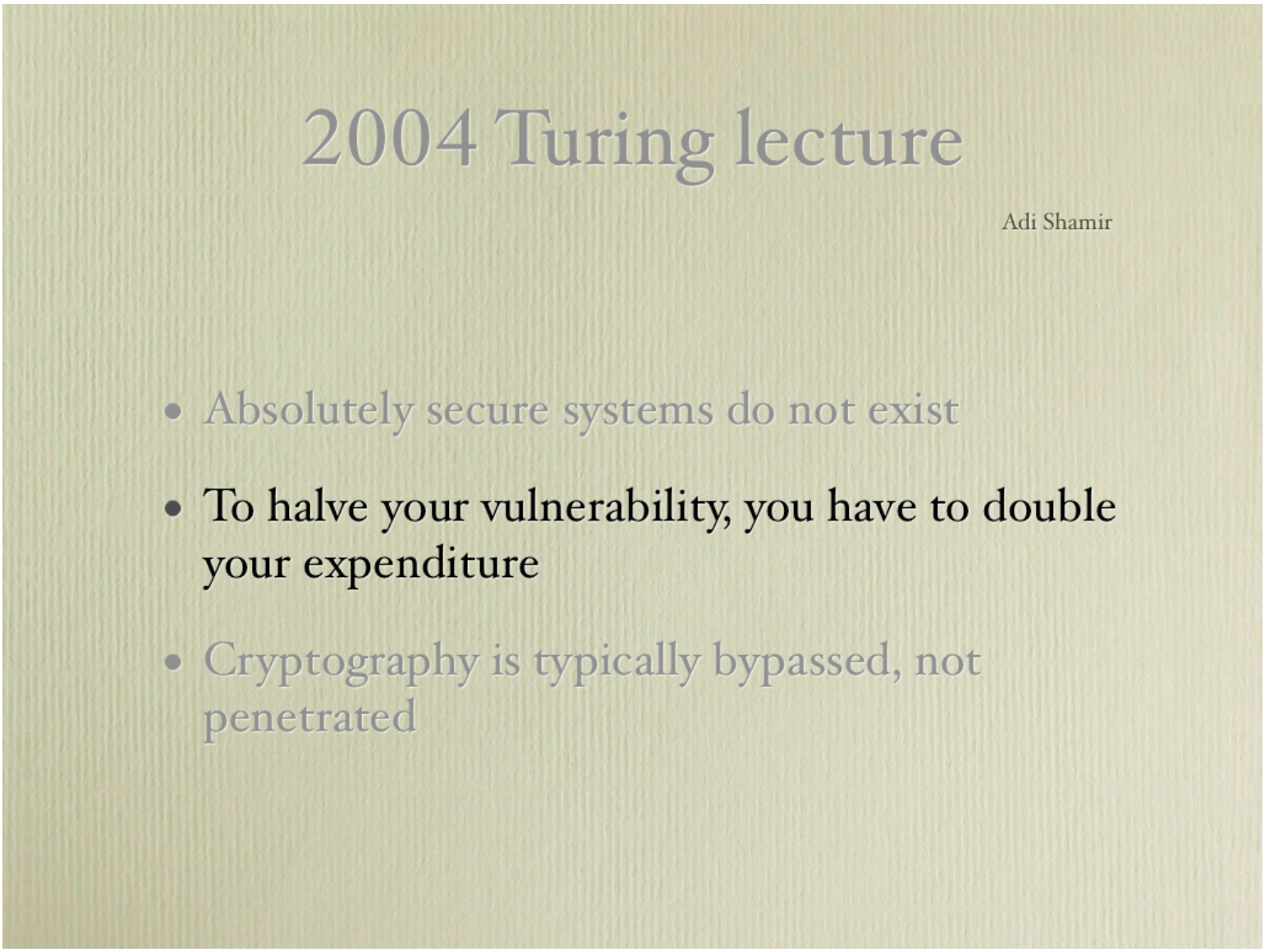
That says that you can get 100% availability two ways: making a system that never fails, or providing a recovery mechanism that is instant. This is, of course, easier said than done as achieving either infinite MTBF or zero MTTR is asymptotically impossible.

There is one point about availability that must be remembered, and that is that the ordinary,belief that redundancy is the answer is naive in the digital sphere.

Redundancy

- If risks are uncorrelated,
  Then redundancy raises Availability
- If risks are correlated and propagable,
  Then redundancy lowers Availability

Our risks are correlated (common vulnerabilities) and propagable (pervasive internetworking). This has consequence with respect to the costs of countermeasures. In his Turing Award Lecture, Adi Shamir gave us a rule of thumb.



The synthesis of these ideas—that security is a subset of reliability, that never failing and instantly bouncing back are each separately unobtainable but collectively valuable, and that to halve your risk you must double your protection expense, add up to a set of risk management choices that include increasing systemic predictability and garnering expertise at handling bad outcomes. I might even go so far as to say that your first concentration might be on the handling of bad outcomes just as a town without a building code needs to first get itself a fire department.

This is true in many fields, and understandable by many from other fields, too. Civil engineers don't want tunnels to collapse, so they design in a margin of safety just like we pick cryptographic key lengths that are in excess of what can be broken by brute force. Cardiologists will put you on medications only after they are well tested, but will encourage you to get regular checkups anyway and, perhaps, to carry some nitroglycerine with you at all times, just as we try to do access control to data but still install egress filters. Lawyers will write contracts intended to help you make money, but those contracts will provide recourse for when things don't go well, just as we write data handling policies but still force trustworthy people to carry only encrypted laptops.

But the natural world is where the truest analogies are, the place where we can learn the most if, for no other reason, it has been around so very much longer than we have.

The rain forests are the oldest biome on earth which is to say they have enjoyed to longest run of stability, unlike, say, the boreal forests of Canada which were under a mile of ice a mere 10,000 years ago. As such, rain forests have also the greatest number of species, which is to say the greatest number of niches and the most complex inter-dependencies. Twenty-five acres of Malaysian rain forest will have more species of trees than the U.S. and Canada combined. Forty-three species of ants were found in a single tree in Peru— the same number as in all of the British Isles. The oldest biome in North America, the temperate rain forest of the southern Appalachians, has more species of lungless salamanders than anywhere else on earth. Yet, the equilibrium of the rain forests is being punctuated by a much more recent invention, Homo sapiens with power machinery. Some estimate species diversity in these rain forests as now draining away at the rate of five species per hour.

The important thing to realize here is that to the biome, to the occupier of a niche that was here yesterday and will be gone tomorrow, this change is a surprise. No tree can say "Time to move" and no salamander can think "I need to evolve." Evolutionary change depends on this unpredictability; otherwise

yesterday's winners are tomorrow's winners, yesterday's dominant species only get more dominant tomorrow.

Note that "change" is not a synonym for "disaster." Some pines have cones that only open after they've been burned in a forest fire. This kind of opportunism for disaster is something we can see in our world—today and every day—in that for every backdoor some worm or virus installs, you can bet your paycheck that there is some other bit of malware that is searching for the self-same backdoor for its own purposes. A backdoor unused is like a biological niche unoccupied; Nature, both biologic and digital, abhors a vacuum. That backdoor will get used, the only question is by whom or what.

Let's illustrate this specifically. Everyone but everyone classifies the 9/11 attack as out-of-nowhere—a black swan to again use Taleb's terminology. That attack changed everything because it was not foreseen. It was a physical attack, but we, here, deal in digital attacks. Many of us have heard the phrase "Digital Pearl Harbor" and many of us here have wished we hadn't. If we talk with a member of the general public, we are likely to hear something like "Look, you paranoid worry-warts keep predicting a big problem and if it was all that likely it would have happened by now. In fact, every day that goes by without something like that happening makes it more likely that it never will. Would you just stop bothering me?"

Now, a statement like "That we have gone this long without anything big happening" is precisely the kind of statement that expects stability to continue, and which is necessary but not sufficient for a punctuation of that stability. If we look at 9/11 as digital security people, we might remember that the NIMDA virus appeared the evening of September 18, 2001, i.e., a week later. Until that point, we'd never seen a virus that had carried more than one method of attack, and NIMDA had five. So, to begin with, even if we had known everything about each of those five methods including population statistics for the numbers and connectivity of vulnerable machines, we would not have predicted

the ability of NIMDA to spread as it did as we had not yet thought to model the union of multiple vulnerabilities.

That, however, is not all. For writers of classic virus attacks, the measure of their success is the energy differential between the first entry into a given target and the second, i.e., the bigger the difference in how hard it is to break in the first time and how easy it is to break in the second time, the bigger the win. The lowest energy way to maximize this energy differential is to install a new backdoor. NIMDA followed this pattern and installed such a backdoor.

Because NIMDA had five methods for propagation and because it was evidently written with speed in mind, NIMDA was also the fastest spreading virus we had yet seen. That rate of spread is known amongst infectious disease people as virulence, and we'll return to that in a moment.

As you know, nearly all malware in the wild persists there. An older virus called E911 was such an example. E911 would cause your modem to dial 911 repeatedly; that is all it did. Now when I call you on the phone, the circuit stays up until the calling party disconnects. When I call 911, however, the circuit stays up until the called party disconnects, a difference that is done at the switch for the obvious reason that you do not want the intruder to cut the phone line and the Police Dispatcher to have to say "Now whom was I talking to?" For the Police to hang up on a 911 call when the calling party has gone away requires a human decision, made under uncertainty, done at human time scales. Because of this, it is possible to saturate a 911 console and that is precisely what the E911 virus was crafted to do—saturate a 911 console.

The E911 virus was old and forgotten on September 18, 2001, but it was still available on the net and, of course, the Internet in the fall of 2001 was still dominated by dial-up connections. We got lucky in the simplest, stupidest, dumb luck kind of way. No jackass had the imagination to grab the E911 virus and re-target it at the backdoor NIMDA was busy installing at warp speed everywhere while we all were pre-occupied with watching CNN 24x7. If someone had done that, then everyone in America would have gotten up the

morning of September 19 only to find that there was no emergency service available nationwide; it would have been turned off everywhere and all at once, like a light switch. While that would not have been a disaster of a physical sort, I submit that it would have been a grand mal seizure of the public confidence. Clinically that defines terror, it would have required no planning just opportunistic reaction, and it would have been an unpredictable event whose downstream influence was out of all proportion to its concrete effects. It would parallel the Treasury's position that money lost or banks folded is a private tragedy of no importance, but that public loss of confidence in the financial system must be avoided at any bearable cost.

On September 18, 2001, we escaped a public loss of confidence by luck and luck alone. As such, the next time someone tells you that the absence of a major Internet attack to date makes the absence of one tomorrow more assured, you can remind them that this proof (that we escaped such an attack by dumb luck) puts to bed any implication that every day without such an attack makes such an attack less likely. It does not make it less likely, but what it does most assuredly do is make it more surprising when it does come.

Note that I am not conjuring up nation state actors or divine intervention, though I personally believe that both are at work and at all times. What I am suggesting is that change is what evolution is about, that change is rarely steady but rather tends to be abrupt, that change is event driven, that the amount of change an event engenders is proportional to the surprise with which that event arrives, and that we cannot make this otherwise. Our preaching on this topic wastes airtime and, which is worse, the more we say "It is coming" the more those who live in the moment will have reason to ignore us.

A moment ago I spoke of the word "virulence." To a physician, virulence is how badly some bacterium makes you cough, sneeze, and worse, but to a bacteriologist, virulence is a innate measure of how good your immune systems is at killing that bacterium since if you can kill it with assurance, to survive that bacterium must cause you to pass it on to the next victim and to do so quickly

enough that your immune system does not cut the skein of its life. Better immune response means greater virulence.

We can thus test a little hypothesis of prediction. Working with bacteriologist Trudy Wassenaar, I plotted the date of appearance and the speed of transmission for the big name computer virus attacks from 1995 on. What we found was just what an evolutionary model would predict—virus appearances became progressively rarer, likely due to progressively competent computer immune systems, but when they did appear each was much, much more virulent than the one before. In other words, the malware called a virus, while perhaps no longer of much interest, showed the pattern of evolution that you would expect in the natural world, only evolving faster in time than random mutation would predict and thus confirming the view that sentient opponents can do no more than make the evolutionary clock run faster.

Virus attacks have, of course, become rarer over time, which is to say that where infectious agents once ruled, today it is parasites. Parasites have no reason to kill their hosts—on the contrary they want their hosts to survive well enough to feed the parasite. A parasite will generally not care to be all that visible, either. The difference between parasitism and symbiosis can be a close call in some settings, and of the folks who famously bragged of being able to take the Internet down in twenty minutes, one has said that a computer may be better managed once it is in a botnet than before since the bot-master will be serious about closing the machine up tight against further penetration and similarly serious about patch management. Therefore, since one can then say that both the machine's nominal owner and the bot master are mutually helped, what we see is evolution from parasite to symbiont in action. According to Margulis and Sagan, "Life did not take over the globe by combat, but by networking." On this basis and others, bot-nets are a life form.

As most of you know, I have argued for some species diversity in those parts of the computing infrastructure that we care to call essential. Stating the obvious, no one calls their part of the computing infrastructure inessential, so we might

as well look at it all. The word you are waiting for me to say is "monoculture" and I've now said it. The diversity of the natural world is something we agree on far more than we disagree, preserving it as an end is a near universal desire and, like computer security, all the contention is around the means, not the end.

But we, laughably unintelligent designers, prefer monocultures whether it is of dry land wheat or corporate desktops. John Evans, of the University of the South, observes that when a natural hardwood forest is replaced with a pulp-wood monoculture "The crop fails in the first rotation, because the beetles go from being a native disturbance to a native epidemic. It only gets worse when you increase their food supply." Remember, if you will, that a patch is an advertisement of where the opponent's next meal is coming from since, as Gerhard Eschelbeck showed, patching behavior is precisely like radioactive decay—in each succeeding interval, half of the then unpatched machines are patched and, in any case, 80% of exploits appear within the first half-life of patch-announced vulnerability and wreak 85% of their damage in their first fortnight.

As there is not a person in this or any room who will argue for explicitly reducing the species diversity of the natural world as a national goal, we have to ask why we do it in the synthetic biomes of enterprise computing. Despite what you might think, I am sympathetic to the actual reason we do it—making everything almost entirely alike is, and remains, our only hope for being able to manage it in a consistent manner. Put differently, when you deploy a computing monoculture you are making a risk management decision: That the downside risk of a black swan event is more tolerable than the downside risk of perpetual inconsistency.

What I am driving at is that the natural world has lessons for us and that we are pretty miserable at intelligent design; provably poorer at it than whatever process brought Nature to us in the state it was in before we began to demolish it.

In the natural world, a high presence of attack pressure must and does result in a high rate of mutation. What part of your body suffers the most daily insults and thus mutates all day, every day? The E. coli in your gut. Their mutation rate rises and falls in relation to stress since if things are going well a mutation is likely to be deleterious whereas if things are going badly it may well be a last chance and, in any case, reproductive fidelity is more metabolically expensive than producing mutations. I suggest that some doctoral student might take a close look at whether a withering digital attack ought to provoke some adaptive mutations in the target. Put differently, if you are losing a game you cannot afford to lose, try changing the rules.

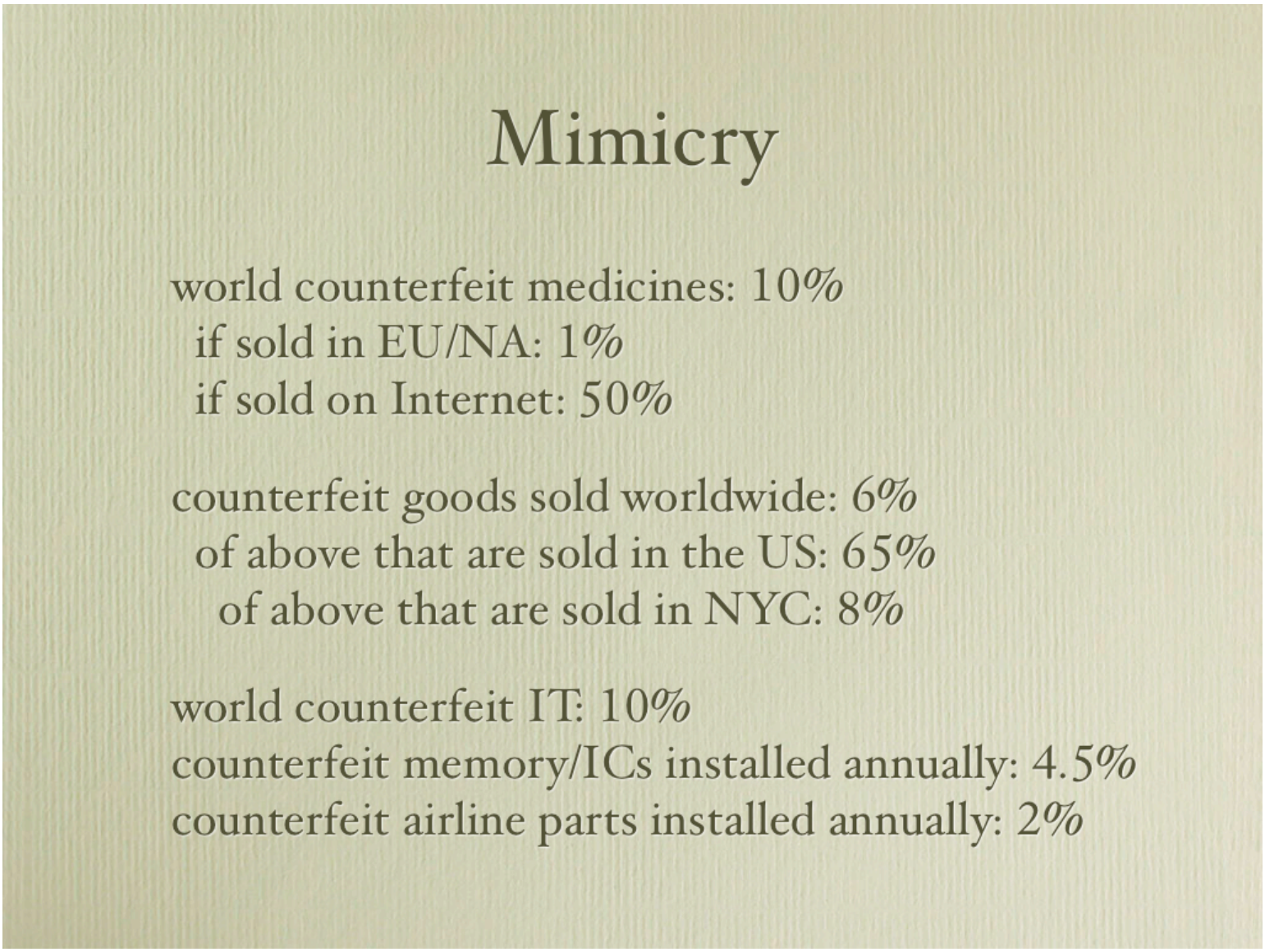
Social insects are the great success story of the natural world; there are 2^50 ants on the planet, the biomass of ants plus termites is 1/3 of the biomass of all terrestrial animals, and honeybees alone confer more economic benefit to humanity than all other insects combined collectively withdraw. I simply do not have time today to go into all the ways that social insects and armies of computers on networks are alike, so I'll just refer you to my paper in ACM Queue from last April. One hint, researchers were able to show with honeybees precisely what I said was true for computer monocultures -- that a hive all genetically alike either wins decisively or fails catastrophically, while a hive genetically diverse neither wins nor loses in any spectacular fashion at all.

In the natural world, a plant that is being eaten alive by, say, aphids will manufacture a come-hither scent that draws ladybugs. If you are a ladybug and you smell that "The Aphid Diner is now open" aroma, what would you do? Bank tellers may well cooperate with the robber, but you can bet they've pressed the button on the floor that calls the gendarmes. We do this with our honeypots— they allow themselves to be taken down by some thug, all the while snapping pictures for the Network Police. In a sense, our intrusion detection systems do the same—they call the police while they are going down.

Honeypots are, if nothing else, a kind of mimicry. Nature has many kinds of mimicry. To mention just two, there is Batesian mimicry which is where some

prey species tries to look like some non-prey species, a sheep in wolf's clothing as it were, and there is Muellerian mimicry where all the ones that, say, taste bad look alike to make up for slow learners amongst the predators. To my knowledge, we don't much have this kind of protective mimicry in the computer field, at least yet. We have the reverse—honeypots try to look tasty and vulnerable when they are not. As such, honeypots are more like angler fish, and, as with everything else today, we may have something to learn from Nature here. Perhaps every network segment needs a honeypot, a model that the common field cricket uses when the male who is singing is targeted by predators but that male will be surrounded by other males who do not sing, only mate.

A kind of mimicry we do have is counterfeit IT equipment, now estimated at 10% of all IT equipment in place. In a few days, I'll have an IEEE Security & Privacy column out on this, but here is a taste:



Put differently, it may soon be the case that the most impactful counterfeiting on the planet is that in the digital arena.

Maybe the answer to all these risks is not to avoid them but to absorb them, what might be called co-existence, or what OASIS in the U.S. or MAFTIA in Europe called "intrusion tolerance." It isn't as if there is any new news in saying that by now we shouldn't have buffer overflows in applications yet, of course, we still do. Perhaps the answer is to just learn to tolerate those overflows and/or the intrusions they tend to enable. Medical science hasn't

solved the common cold, either; we've just learned to tolerate it after effectively giving up on immunization as a public health strategy.

There are at least 350,000 species of beetles, which led to the biologist Haldane's remark that "The Creator, if He exists, has an inordinate fondness for beetles." This degree of speciation means beetles have an authentication problem, viz., "With whom can I mate?" We do authentication using cryptographic locks and keys where neither keys nor locks are interchangeable. Beetles have the analog equivalent; the mechanics of the male's penis will only match the mechanics of a female of the same species, enough so that properly curated specimens in museum collections will exhibit the penis adjacent to, but detached from, the rest of the specimen. One can stretch and say that the various mating dances of beetles and higher forms are just so much pre-authentication.

While some people like to say "Specialization is for insects," tell me that the security field itself is not specializing. We have people who are expert in forensics on specific operating system localizations, expert in setting up intrusion response, expert in analyzing large sets of firewall rules using non-trivial set theory, expert in designing egress filters for universities that have no ingress filters, expert in steganographically watermarking binaries, and so forth. Generalists are becoming rare, and they are being replaced by specialists. This is speciation in action, and the narrowing of niches. In rough numbers, there are somewhere close to 5,000 various technical certifications you can get in the computer field, and the number of them is growing thus proving the conjecture of specialization and speciation is not just for insects and it will not stop.

Lest you think that it is too far fetched to consider a computer a life form, subject to evolution just like any other life form, consider embedded systems. They are already two orders of magnitude more numerous than keyboards and displays hence the future threat space, which we must lead in the same way one leads the deer when hunting, is a threat space where a computer is not identifiable as such, but is instead inside some nondescript appliance. Only this

week it became widely known that China will soon require source code to digital devices sold or made there.

So should or should not an embedded system have a remote management interface? If it does not, then a late discovered flaw cannot be fixed without visiting all the embedded systems which is likely to be infeasible both because some will be where you cannot go and there will be too many of them anyway. If it does have a remote management interface, the opponent of skill focuses on that and, once a break is achieved, will use those self-same management functions to ensure that not only does he retain control over the long interval but, as well, you will be unlikely to know that he is there.

This leads to a proposal on what to do about the future: Embedded systems, if having no remote management interface and thus out of reach, are a life form and as the purpose of life is to end, an embedded system without a remote management interface must be so designed as to be certain to die no later than some fixed time. Conversely, an embedded system with a remote management interface must be sufficiently self-protecting that it is capable of refusing a command.

I define complexity as the density of feedback loops. A lot of people say that complexity is the enemy of security—I'm one of them—but at the same time I am here to argue that we have to learn from Nature precisely because Nature is the most complex thing we will ever see. Nature is an existence proof that complexity is not the enemy of life, but complexity is the enemy of stasis. Our problem is that we've pretty much equated security with stasis, and it is slowly getting us into trouble. In the financial world, like in our world, there is zero doubt that we humans can build a system more complex than we can understand. The same is true in the digital world.

We today have a financial system that is global, interconnected, complex, and more prone to cascade failure than when it was not global, when it was not interconnected, and when it was less complex. You either save (patch) the existing system, or you start over.

The parallels to our security world are real and precise:

finance	*digital*
monolithic	monocultural
globalized	internetworked
transaction triggers	SOA libraries
derivatives	code re-use
meltdown	BSOD
bail out	wipe/re-install

And again, this is true in Nature. Take forest fires—if you always quench them, such as to protect vacation homes and tourist dollars, then you necessarily build up the supply of unburned fuel wood in the ecosystem and someday you get a much bigger fire. If you let any and every fire burn, someone who can vote will lose. If you prevent any and every fire, you look smart and life goes on, and predictably so... until it doesn't.

If we look at Nature in the form of the equations of ecology, we also see two alternative games for survival, r-selection and K-selection. R-selected species produce many offspring, each of whom has a relatively low probability of surviving to adulthood. By contrast, K-selected species are strong competitors in crowded niches, and invest more heavily in much fewer offspring, each of whom has a relatively high probability of surviving to adulthood. If we change the term from "produce many offspring" to "re-image frequently" you now have precisely the advice Microsoft's D'Anseglio gave when he said, "[In] dealing with rootkits and advanced spyware programs, the only solution is to rebuild from scratch. In some cases, there really is no way to recover without nuking the systems from orbit." This brilliant remark is a direct, if inadvertent, suggestion that desktop machines need to be r-selected, i.e., they need to die and be re-born often. If you are of a mind to invest in virtual machines, you may get r-selection as a side effect to whatever it is that you are trying to do with VMs.

My bacteriologist mentor taught me a further thing. The higher up the evolutionary ladder you go, the greater the percentage of the organism's total metabolic investment goes to self protection, topping out with humankind's investment of twenty years to raise its children. Her thought, and now mine, is that since it is abundantly obvious that except for people in this room, there are few computer users who are doubling their work output every eighteen months, hence the dividend of Moore's Law should be spent, in increasing fractions, on protecting the computer and the data it holds. My previous company's product effectively suggests that you spend some of the 82,000X dividend Moore's Law has provided since the 1983 publication of The Orange Book on implementing the Reference Monitor, to take an example close to (my) home.

I already spoke of one equilibrium punctuation, that of the creation of our industry as a side effect of adding a TCP/IP stack to Windows. There has been another, and there is about to be a third.

The second of these moments occurred, as far as I can tell, some time around 24 months ago. Like the first, it was no thunderbolt, more like a glacier finishing its slide across a river bed and thus "suddenly" damming the waters. This moment was when our principal opponents changed over from adventurers and braggarts to professionals. In a sense, professionalization of the attack class is akin to virulence in that the increasing immunity of computer systems forced an upgrade in the ability of the attacker to attack, i.e., finding vulnerabilities and exploiting them is now hard enough that it has moved out of the realm of being a hobby and into the realm of being a job. This changes several things, notably that hobbyists share their findings and are paid in bragging rights whereas professionals do not share and are paid in something more substantial than fame. Speaking biologically, a mutation (toward strength) on the part of the prey was matched by a mutation (also toward strength) on the part of the predator. As a side effect, the percentage of all vulnerabilities that are unknown has risen and will continue to rise. We have yet to reach the post-punctuation equilibrium.

This mutation toward strength represented by professionalization of the attack class was not a simple, compensating match for the increasing self-protection in merchant operating systems. It went further and it did so because, at least for the first world, the digital arena is now clearly where the opportunities are, such as that when robbing banks it is the amateur who uses a hand gun and the professional who uses a bot.

In the fall of 2006, I did some back of the envelope calculations that resulted in a guess that 15-30% of all desktops had some degree of external control present. I got a bit of hate mail over that, but in the intervening months Cerf said 20-40%, Microsoft said 2/3rds, PC Tools said 70%, and IDG said 3/4ths. It doesn't matter which is right; what matters is that this changes a core feature of the ecosystem—and changing a core feature is the very definition of a punctuating event. In this case, it actually was not standing up a professional class of attackers any more than in the first go 'round it was a spike in the

second derivative of the reported attack rate. What it was was that a fundamental assumption of network security has now been breached and there is no putting it back together again.

Ever since we did Kerberos, the idea has been "I'm OK and you're OK, but the big bad network in between us cannot be trusted for a second." Authentication, authorization, and accountability all begin with authentication and that, in turn, begins by asking the Operating System the name of the user. What has really changed is that it is not true that "I'm OK and you're OK" since it is entirely likely that the counterparty to whom you are connecting is already compromised. A secure network connection? Who cares if the other end is hosed. Gene Spafford was right but early when he likened network security as hiring an armored car to deliver gold bars from someone living in a cardboard box to someone sleeping on a park bench.

That is the new security situation you and we are facing—what to do about 0wned counterparties. This is a today issue, not a tomorrow issue; the November 2006 10-Q filing for E-Trade included a material loss due to exactly this problem, the first SEC filing of this sort to my knowledge. 0wned machines mean key loggers and key loggers mean opponents who can get you to help them in the pump phase of a pump & dump stock fraud, whether you like it or not. If and when you ever bother to call your discount broker to complain that this or that purchase was not one you did, the broker has two choices: "You are an idiot." or "We'll make it up to you." Such a situation is untenable.

The most likely option, and it is being quietly implemented in several places today, is for some kinds of transactions to be based on the merchant side 0wning the customer side for the duration of the transaction. Whether this comes as an Active-X control, some sort of use-once browser, or what remains to be seen. In clinical trials, pharmaceutical companies long ago found it was safer and easier to manage if they just shipped a laptop to the participating doctor. Brokerages probably won't do that and random e-commerce merchants

absolutely will not, so we are back to whether it is a good idea for the merchant side of the transaction to assume that the client side is already compromised and to compromise it for a moment on its own account. This is no easy decision for either side, but if the average customer's choice is a no-loss guarantee in exchange for a moment of remote control, then it is my bet that they will take the offer. Whether this is how it goes or not is, however, irrelevant—I am just using this as an example of an adaptive reaction to our opponents becoming good enough that the original "I'm OK; You're OK" starting point for network security no longer applies, and some mutation will have to replace it.

By the way, if you think the professionals aren't winning, just consider that they now value stealth over persistence, i.e., they find it so easy to 0wn machines that they make no effort to survive reboot, preferring instead to hide in-core only.

I've been talking fast and, possibly, in circles. I am at that stage in my life where I can truthfully say that the more I know, the more I am aware that I don't know. The unknown unknowns are either getting bigger for real, or they sure look that way from where I sit. I think they are getting bigger, and I think complexity is the reason why—in a complex world there are more places for unknown unknowns to hide, and there are therefore more of them in the same way that complexity in the digital world, like complexity in the natural world, means new niches all the time.

I am certain that digital insecurity is our fault, but I am ever more skeptical that solving it doesn't require a crash or letting the forest burn. I don't mean this as a pessimistic thing—I mean it as trying to learn from that which was here before we were and will be here after we are gone. I mean it as someone who believes that numbers and measurement and inference and engineering discipline got us out of caves to the amazing comfort we now enjoy. I also

mean it as saying that we are halfway to the moon and it is time for a mid-course correction, only the harder I look the more challenging I find the idea of just making a tweak here or there.

I've left out the possibility of cyber security becoming part of warfare, though it surely will. I've concentrated on the big picture of how it is that selection works, how important is the role of great numbers of small random changes versus the time it takes for those to make a difference. I am trying to learn from the greatest teacher, the sum of the natural world we've been given and the vast time it has had to try ideas we don't have the nerve or the time to try. I bow my head in reverence to what I have come to believe is a world too complicated and beautiful to have come to be just by chance, but rather a world that, at this point in history, we have no choice but to intelligently design or fail.

For you just starting your career, you have even more scope that I did and I hope you see it that way. For those of you with the better part of a career behind you, I invite you to imagine that the computer world is already a life form, and to ponder on what that means.

There is never enough time. Thank you for yours.