The Price of Belief: Insuring Credible Trust? (Transcript of Discussion)

Bruce Christianson(✉)

University of Hertfordshire, Hatfield, UK
b.christianson@herts.ac.uk

I'm going to endorse several of the things other people have already said. The basic thesis of this talk is that we are getting better and better at solving the access control problems we had in the 1970s, and it's probably time to stop doing that and start thinking about the problems we've actually got now.

We're currently in the position of these politicians saying, "Okay, perhaps we need to tweak our policies, but the real problem is that the electorate just aren't getting our message." The electorate are saying, "Actually the message is coming across loud and clear, and we don't like it. We want you to come up with some different policies that we might actually vote for."

We're trying to sell a vision of security to users that they simply won't have any part of. It's not that we need to work out a slightly better interface for presenting our current mechanisms to users. They are just not going to act the way our models require, and we need to adjust our world-view to take account of that.

One of the differences between our world-view and theirs, for example, is that we still see security as absolute. A system is secure or it's not. Somebody is authenticated or they aren't. They are either authorized to do something or they're not. In the real world, it isn't like that. People don't have binary beliefs. They have views, and they are happy to make decisions based on their views, but those views are then going to change depending on the outcomes of their decisions.

If Dr. X wants to do something on Ward 12 then okay, some things are clear-cut. No, she definitely *can't* authorize a payment of a million pounds; yes, she definitely *can* look at this particular patient record. But there's a lot of stuff that's in a grey area in the middle: sometimes it's not so much that she is not *authorized* to do something, it's more that I'm not *expecting* her to *want* to do that. *Why* does she want to do that?

In some cases I'm quite happy for her to do it, but I really want a much higher level of authentication than I would do for other things. However it's probably not helpful to to interrupt her, right at the moment that she's engaged in a crucially sensitive procedure, and say, "To proceed, you must now enter the authentication code showing on the device that you've left in your handbag which is in your office on the next floor." It's probably much better if the system pages a staff nurse and says, "Can you verify to me that Dr. X is in the ward right now, and also can you tell me what she appears to be doing, please?"

In the real world we're inclined to decide whether or not something's okay based on previous dealings that we've had. We're probably all right about saying,

J. Anderson et al. (Eds.): Security Protocols XXIV, LNCS 10368, pp. 39–46, 2017.
DOI: 10.1007/978-3-319-62033-6_6

"Okay. So you forgot your identity card, can you get someone that we *can* authenticate to verify who you are to us please?" Yes, they might be lying, but we're primarily trying to reduce the *risk* of making a bad decision. This interacts with the problem touched on a little bit earlier, which is that we're outsourcing security. More and more access control decisions are being made by a third party which doesn't have an investment in the client, or in the data owner.

This kind of outsourced access control mechanism naturally desires to have very binary decision trees that lead to a very clear-cut audit trail. This means they can show that the patient died, the plane crashed, and the money was embezzled; but it wasn't their fault, because they can prove that they complied with the policy that they were given. The difficulty is we're outsourcing security, but we're not delegating the risks and opportunities in a way that aligns the incentives. So the incentives are not aligned: the "good guys" don't all want the same outcome anymore, no matter what the security model says.

For example, denying access is no longer the "safe" option. It's not true to say, "Nobody ever got fired for denying access in a medical scenario," for instance. Interestingly, in the medical scenario, we do now see new protocols emerging, such as the break-the-glass access control protocol. There's no actual glass involved, by the way, it's purely a noughts-and-ones protocol, that allows non-binary outcomes to access control decisions. It's not just deny or allow, there is the option to say "This is a special case that we didn't see coming. We need to allow this for now, and shall sort out afterwards whether we should have allowed it or not." Access control mechanisms need to allow explicitly for these tradeoffs.

The model I'm going to use to do this is the insurance model. I'm not actually suggesting that we need to quantify risks in the same way that insurance adjusters do. Although, I suppose you could do that if you want. Alice could charge Bob a cost for doing something that is based on what Alice expects it to cost her to let Bob do it. But actually it's enough if Alice picks the amount she charges Bob to be such that it aligns his incentives with hers. The cost is designed to promote wholesome behaviour over unwholesome behaviour on Bob's part, rather than to make Alice a "profit".

The idea is to look at who is risking what. We've got a server that's going to decide whether to allow access or deny it. At the moment, if the server allows access when it shouldn't, the server's going to suffer. But, consider the person with the rights on the data object: if nobody accesses that object, they're going to go out of business. So default-deny isn't acceptable either.

It's a little bit like the situation with shoplifting. If you own a shop then you have to put up with a certain amount of shoplifting. It's easy to eliminate shoplifting completely, but then no one will shop in your shop. If they're getting their crotch sniffed by a large dog every time they go in to buy a can of tomatoes, they'll probably shop somewhere else.

So, it's a matter of presenting users and data owners with the incentives and the disadvantages, the contextual clues that we use in real life, and allowing them to make, possibly in an automated way, rational decisions based on that information.

The first thing is to get straight about what are the risks, and what are the opportunity costs? This isn't new, David Wheeler was advocating a version of this approach in the fourth of these workshops 20 years ago. However I think David tended to underplay the fact that variance was important as well as expectation: very often you do want to give people a way to trade one off against the other. People buy insurance, but people also buy lottery tickets. In each case, they are willing to accept a less favourable expectation in exchange for either a higher or a lower variance.

The proposal that we're making is to apply this approach to security decisions. Let's think about the risks and opportunity costs, and then let's pass them along. The data owner might say to the server, "I'm happy for you to grant this access provided you obtain this stipulated amount for it." You can think of this amount as being like an insurance premium. And the server might choose to pass that cost along to the client, if that's the model you want to use. It might be that the system says to you, "We're not sure whether you are Dr. X or not, but I'll tell you what, if you put 1000 pounds into the machine in the corner then we'll give you access. And maybe you are really a journalist, but at least we got 1000 pounds for the data."

If Dr. X subsequently logs in, authenticates herself, and validates the transaction, then we'll give her your 1000 pounds back. Or maybe we want two people that are authenticated to each put 100 pounds in and say that you are Dr. X. If Dr. X subsequently authenticates herself "properly" and affirms the earlier transaction then the two guarantors get their money back. Or maybe clients are going to pay in terms of reputation. Perhaps there's a kind of a credit score, or loyalty points. Frequent flyer miles. Something like that, that you can spend in order to obtain these accesses. And maybe how *many* points you require for an access depends on how contentious that particular access is.

We might say, "Well, if it turns out that the access was bad and you shouldn't have had it, we'll charge you a million points. Alternatively, you can pay 100 points up front now, and we'll indemnify you against this going wrong." Just like an insurance policy. Or else we might have a system that's more like a bail bond, where you buy the bail bond; we let you into the system; but if it turns out that we shouldn't have, then we send the equivalent of bounty hunters after you.

The user interface for our new approach would be very different to the sort of pop-up box you typically get at present: "There is a security problem. The certificate from <`some server that you have never heard of`> is self-signed. Would you like to 1. Proceed? (In which case whatever goes wrong, it's your fault.) 2. Abort? (In which case there is no way for you to get what you want done.) or 3. View the certificate?"

I love that third alternative; I think this is an absolute brainwave on the behalf of whoever thought it up. It's a totally brilliant idea to have a non-expert user confronted with an endless string of ASCII or if they are lucky X.509 or something. They can look at that for a while and then they have to click on one of the other two buttons anyway.

Instead of all that, our approach would have a pop-up box that said, "What you are trying to do is an unsafe sort of transaction, and here are a number of options. You can risk a million Tiger points; or you can pay a hundred Tiger points now and indemnify yourself against it. Or you can take the time and trouble to authenticate yourself and various other parties more carefully, in which case the price will come down to about 8 points, depending on how much effort you are willing to put in." Or you can have various other options, which are based on how carefully you've authenticated yourself, what kind of protocol you're connected over, how many of your transactions have gone bad in the past. The key point is that we do not expect you to have a fetish about getting perfect security. The point is rather that you're on a limited budget. Your supply of these loyalty tokens, or whatever it is, is limited.

Ross Anderson: This is perhaps very apposite given all the discussion about exceptional access. If you want exceptional access as a civilian, either with an Anton Piller Order or Norwich Pharmacal Order, you have to give money up front to lawyers. For an Anton Piller Order you typically put twenty or thirty thousand pounds surety, in case the person whose house you search comes back and sues you for damages. If the UK police are looking to get information from a US service provider like Google or Yahoo, they typically have to go through MLAT, which involves time and expense. All of these things are good. The fact that it used to cost them 200 pounds to get a mobile phone location trace was good, because it meant they didn't do it all the time.

The problem is that governments try to legislate for zero marginal cost access. How can you prevent this systems failure with this insurance structure?

Reply: In other words, how can we prevent governments from requiring us to insure them for nothing? I don't have a good answer to that, we're primarily concerned here with the commercial world. It's really hard to throttle governments against their will, but we can try to persuade them that it would be in their own interest to limit the rate at which they are able to do things. Chelsea Manning is an example of somebody doing something that they were authorized to do. Do you want to allow a sysadmin to move a file from once place or another? Yes, you do. Do you want them to be allowed to do it four million times in quick succession? Probably not. Associating a very small cost with each time they did it, that counts against an allowance, would allow you to detect and respond to that quite rapidly. The government is perhaps missing a trick here by not applying this type of throttle to its own security policies, internal as well as external.

The key point is to give the user an interface where they feel they're in control. They're presented with some data. They're making a decision. We've rigged the game so that their incentives are aligned with ours as security people, and we're giving them a budget and more importantly we're expecting them to spend it. Instead of saying, our objective is perfect security, we're saying, our objective is security that's good enough against some metric.

Jonathan Anderson: There are some other currencies, which you haven't talked about, which involve functionality and performance. We might say, so,

your certificate appears valid and it's signed by a CA who's in our list of trusted CAs; however, that's a very long list and our certificate transparency doesn't really like it so, tell you what, we'll let you view the webpage. We're not going to ask you whether you *really* want to view the page that you already said you wanted to view, answer yes, or no. But we won't run JavaScript. Or we won't display password prompts. Or there's some degradation of functionality. Or we'll have increased sandboxing that makes the thing run much more slowly, but ... [next slide] What a very nice slide you have there [laughter].

Reply: And what a very insightful comment. Okay. The other basic problem with access control is that currently outcomes are binary. It's allow or deny. There's nothing in the middle. But, just as Jon points out, we could allow the access but put you in some sort of sandbox, where you can't do anything terribly irrevocable. It's fine to explore the catalogue, but we're not actually going to let you buy anything. Or you're going to be subject to audit controls, perhaps definitely or perhaps you're just going to have a higher chance of being subjected to an audit control.

A very small proportion of transactions are routinely selected for a random in-depth audit. There's always a problem with people gaming to see where the threshold is for triggering an automatic audit, and then putting through transactions that are just below that level. One of the other difficulties with current access control mechanisms is that we expect them to be deterministic: to give us the same answer if we ask the same question. A good counter-measure may be to have non-deterministic mechanisms, so that whether you are audited or not is actually random, but what you're doing affects the probability.

In summary: non-deterministic algorithms; a premium depending on what access you're requesting, what authentication you're offering, and what your past history is; and choosing from a range of alternative premiums, depending on what precautions you are willing to take. You can decide how much security you are willing to apply and the system can respond accordingly. The access decisions include other alternatives besides allow or deny.

As well as delivering more flexible services and interfaces that users might actually be willing to use, this approach also allows you to do security auditing by simply following the money. For example, if your computer has been taken over and is being used as part of a bot-net, you're going to notice very rapidly that there's a flow of security points out of your system with no corresponding flow of goodness in, and you're going to say, "Why is that? Why am I spending all this stuff and not getting anything for it?"

Simon Foley: Would this work for Acceptable Use Policies? Currently, we're using servers with Acceptable Use pop-ups that you have to agree with in order to get the service. If the conditions are reasonable, you might agree, but if it's totally clueless you might still agree, because ...

Reply: The alternative is not getting the service at all, under any conditions.

Simon Foley: In principle I can say, "Well this is restrictive, or co-optive to my data", but the difficulty is there's no fairness, because once they have my

data I have to take them to court to prosecute, and I can't afford to do that as an individual. Do you think a mechanism like the one you are proposing could help?

Reply: It depends to some extent on what market pressures produce. In principle, it would allow things to evolve that say, "All right, would you instead be prepared to agree to this less onerous agreement in exchange for more limited access?" This is what a lot of academic licenses already do. A lot of data repositories have a licence where they say, if you log on at a university that's one of our clients, then you get access that is restricted, in exchange for not agreeing to some of our commercial conditions.

Simon Foley: Again, there's an asymmetry to it. I might end up agreeing to the policy because I don't understand it, or I have some malicious reaction to the policy and I know I can't be enforced. It comes down to the consequences. The cost to me to demonstrate the uniqueness of that policy is too high.

Reply: I'm going to try and wriggle out of this, by saying I think now you're presenting a softer version of Ross' objection. Ross is the extreme end where you've already signed up involuntarily to a policy that's completely outrageous, and even going to court won't help you. You're putting forward a softer version of this, and asking how far can an approach like mine get you? I think the answer is, if people are doing this for money - if people have a data asset they're trying to make money from - then it's a marketing question. Can I make enough money from suspicious people like you to make it worth my while offering a softened variant of the product?

The answer to this question is not obvious: the reason there isn't a really secure iPhone is there's no market for it, right?

Hugo Jonker: Two comments. One is that you seem to be assuming cloud forums where everyone chooses the system. Like the nice example that you had, where they self-sign certificates.

Reply: No, no, I'm not necessarily assuming that.

Hugo Jonker: The second question is: take the self-signed certificate. How would you determine a good pricing strategy? Imagine I want to attack you. I set up a website and I can find everyone here a deal, bringing down costs. Making me seem very reliable. As soon as you go there ...

Reply: Suddenly I get a much worse deal.

Hugo Jonker: You get a worse deal, with seemingly little risk to me, so it's a spear attack.

Reply: Attacks like that generally work where somebody builds up their reputation because they want to do one big scam. Like borrowing lots of small amounts of money from a bank to build up your credit record so that they'll lend you a big amount of money, and then running away.

Hugo Jonker: This system you propose seems particularly vulnerable to that sort of attack.

Reply: Yes, it is. But if I'm in the position of the bank, which is the position you're putting me in, then that's a risk that I, the bank, am willing to bet on because, in the long run, it works out for me. Okay, I got scammed by you, but by engaging in that type of transaction I come out ahead across the piste in the long run. I expect to lose occasionally, and I'll probably insure myself against that.

Hugo Jonker: Yes, but there are actually two parties here. In the case of the self-signed certificates, the server giving the self-signed certificate and the user accepting it should both somehow be involved in saying, "I accept this risk". Both should somehow put credits into a pot.

Reply: What happens in practice, in the model I'm advocating, is that the server incurs the risk and then may decide to pass it along to the user; either at face value, or with a discount, or with a premium, depending on their risk model.

Hugo Jonker: Then how does the user forward *their* risk to the server?

Reply: Okay, that's fair enough. When I say user and server, this is unsatisfactory if we are really in a peer-to-peer setting. In that case we are talking about arbitrage, we're talking about using pop-up boxes, or whatever, to negotiate a contract.

Ross Anderson: Perhaps there's a simpler approach to this. If one imposed a rule that personal information could not be sold at a marginal price of zero, that might fix many things, because where companies monetize something they won't give it away. If you get access to stuff as a researcher that's also being sold to commercial companies, it comes with an NDA even if it isn't sensitive.

Reply: For revenue protection purposes.

Ross Anderson: Much of the privacy failure is because the marginal price of information tends towards zero for the usual information economics reasons, in the absence of something stopping it. The price of software can be kept above zero by copyright licensing mechanisms. Perhaps what is needed here is a privacy licensing mechanism.

Reply: That would impose a similar lower bound.

Ross Anderson: Which effectively imposes a government tax. Suppose this is the way forward: the Chancellor of the Exchequer simply says that every record containing personal information attracts a tax of one penny, and then all privacy violators are tax evaders and go straight to jail.

Reply: Actually, Caspar Bowden and I once had this very conversation as part of a discussion with the Information Commissioner about how to protect against

information breaches. We reckoned that having a flat charge for personal information would be a very effective mechanism[1].

Jonathan Anderson: I think one of the problems with implementing this model is that the people tasked with enforcing the security policies in most organizations come from a part of the organization that is absolutely risk intolerant. They share in none of the benefits of enabling things, but they get egg all over their face whenever something goes wrong. I think the same is true in accounting departments or typical security in a lot of organizations, but there's kind of a fundamental organizational behavioral problem. How do you fix that? How do you get them to *want* to do this?

Reply: So, how do we get people like us to buy into a model that says, if you're never making mistakes, then you're not taking enough risks; and that means the business is losing money relative to our competition, and that's why I'm firing you. How do we entice security to move into that model? That's a really good question on which to end, I think.

[1] In the scheme Caspar and I came up with, the tax took the form of a per-datum spot fine for being in possession of personal information that was not tagged with a valid ACL and a conforming audit trail.