Vintage Bit Cryptography

B. Christianson and A. Shafarenko

University of Hertfordshire

We propose to use a Random High-Rate Binary (RHRB) stream for the purpose of key distribution. The idea is as follows. Assume availability of a high-rate (terabits per second) broadcaster sending random content. Members of the key group (e.g. {Alice, Bob}) share a weak secret (at least 60 bits) and use it to make a selection of bits from the RHRB stream at an extremely low rate (1 bit out of 10^{16} to 10^{18}). By the time that a strong key of reasonable size has been collected (1,000 bits), an enormous amount of data has been broadcast (10^{19}–10^{21} bits). This is 10^6 to 10^8 times current hard drive capacity, which makes it infeasible for the interceptor (Eve) to store the stream for subsequent cryptanalysis, which is what the interceptor would have to do in the absence of the shared secret. Alternatively Eve could record the selection of bits that correspond to every value of the weak shared secret, which under the above assumptions requires the same or greater amount of storage *i.e.* $2^{60} \times 10^3$. The members of the key group have no need to capture the whole stream, but store only the tiny part of it that is the key. Effectively this allows a pseudo-random sequence generated from a weak key to be leveraged up into a strong genuinely random key.

The stream observation time given a 10Tbit/sec broadcast rate is only 10^6 to 10^8 seconds, or a week to a few months. Over this time the shared secret is not used for any kind of communication and so the only possible threat is insufficient key storage security, which is present in any cryptographic scheme. It is interesting that in our approach the passage of time strengthens the resulting key: the longer we wait before the key is used, the less chance there is that any relevant part of the stream is present in a storage facility anywhere in the world, due to the sheer mass of data. This is, in a way, opposite to the standard assumption of cryptographic strength, that keys becomes weaker with time. Accordingly, we call this system Vintage Bit Cryptography.

It is interesting to note that vintage bits are not a hostage to future technology development: the ability to record more data per unit cost in future has no influence over the present time: vintage bits not recorded now will not become available later. Nor does leaking the weak secret compromise vintage bits obtained earlier, provided the time difference is sufficient to overwhelm the capacity of attacker's stream storage. In particular, schemes such as EKE [2, 4] can be used to leverage the initial weak secret into a strong pseudo-random seed without fear that subsequent development of quantum computers (allowing the easy solution of discrete logarithm puzzles) will expose previously obtained vintage bit keys.

Beacon systems have been proposed before [9, 12, 10], particularly in connection with satellites [13]. A traditional beacon implementation based upon a

geostationary satellite would make the key distribution system available over a wide area at a very small cost to a consumer. But at present digital broadcast satellites lag far behind optical fibre in terms of bandwidth, transmitting only on the order of 10Gbits/sec, although this rate will increase with the use of higher microwave bands.

A satellite solution which could prove more interesting is a swarm of micro-satellites in a Low-Earth-Orbit (LEO). Such satellites could be equipped with an array of tuned silicon lasers that transmit on a number of wavelengths, and with physical random bit generators that control the lasers. Importantly, no radiation protection is required in this case. Indeed, the spacecraft need not have any processing power since all it broadcasts is random digital noise. LEO satellites could be tiny: less than a cubic decimeter undeployed size, with a small production and deployment cost: space scree (rather than dust).

Anyone with a few tens of thousands of dollars to spare can already have micro-satellites launched using a non-governmental space operator. These satellites can keep orbit for years without thrusters and can maintain their orientation by purely passive means. The overhead passage for one of these craft would last 20-30 min, so a continuous RHRB stream at terabit rates would require a hundred spacecraft or so. Using a polar orbit one can ensure that the continuous stream is available anywhere on the planet, and that the area of consistent observation (where all ground observers can see the same satellites at the same time), is of the order of 1000km across, which makes it quite suitable for European applications in particular. The XORing of streams produced from several satellites launched by mutually distrusting parties eliminates the need to trust any individual craft.

However optical fibres are an attractive alternative to satellites, and our primary interest in this paper is with very high bandwidth fibre-optic beacon systems. The first implementation issue to consider is feasibility.

A single optical fibre can already carry more than 1Tbit/sec with a bit-error rate (BER) better than 10^{-3} using an appropriate combination of Wavelength Division Multiplexing and Optical Time Division Multiplexing. Low BER is a key goal of conventional fibre optic communications, but this very tough restriction is not an issue for us. Transmission errors are easily mitigated against by using a simple protocol based on FEC and cryptographic hash functions:

$$A \longrightarrow B : P|Q$$

where $K = K_1|K_2|K_3$ are the vintage bits recorded by A: K_1 is the eventual shared secret with B, K_2 and K_3 are used as one-time pads;
h is a strong hash function, $P = K_2 \oplus h(K_1)$;
and F is a forward error correction function, $Q = K_3 \oplus F(K_1|K_2)$.

The protocol succeeds if B's calculated value for $h(K_1)$ based on the value of $K_1|K_2$ recovered from Q agrees exactly with the value for $h(K_1)$ recovered from P. Note that the message $P|Q$ can be sent over any open, moderately non-lossy channel: no endpoint authentication is required, and data integrity is an issue only if we are concerned with denial of service attacks. In particular, if the message is broadcast, the identity of Bob need not be revealed.

Because low BER is not a consideration for vintage bit cryptography, we are able to propose the use of cheap optical fibre technology which is not suited to the mainstream communication industry. This provides an attractive (cheap!) alternative to the optical fibre systems already being used for key distribution in industry, which use very low bit rates and quantum technology. These quantum-based systems make eavesdropping detectable, but come at a very high cost [6–8, 14–16]. This form of quantum technology also depends crucially on the physical integrity of the optical cable: it eliminates passive eavesdropping but avoiding the man-in-the-middle attack requires at least a weak form of end-to-end authentication for the side-channel, which imposes constraints similar to the initial sharing of a weak secret in our proposal.

Ensuring the integrity of the communication path from a shared beacon is problematic with fibre-optic technology (in contrast with satellites). One simple possibility in the case of a point-to-point link is to co-locate the beacon with one of the participants (say Alice), as may be done in the quantum key agreement scenario. However a more interesting case is where we wish a single beacon on a fibre optic loop to be shared by all the loop nodes. In this case we would like to reduce the integrity requirement to reliance merely the integrity of the beacon itself, and not that of the fibre optic medium.

One possibility in this case is for clients to pre-share a weak secret with the beacon (or more accurately with a co-located trusted server). As they collect vintage bits to share with each other, Alice or Bob uses this weak secret to generate bits shared with the beacon service, over the same observation period and using the same protocol. The protocol between Alice and Bob now succeeds only if the vintage bits shared with the beacon have not been tampered with: if they are correct then the real beacon is the source of the bits shared between Alice and Bob. Otherwise the bits are corked and should not be used. The bits shared with the beacon can be discarded, or used to update the weak secret shared with the beacon. Optionally, the beacon service, since it is trusted anyway, can be used to share an initial secret between Alice and Bob in case they have not already been introduced.

However it may be a disadvantage for a beacon protocol to require per-client state to be kept at the server end, and individual communication between each node and the server along the side channel. An alternative is to use a variation of a Merkle-type protocol [3], combined with an additional lower-bandwidth authenticated broadcast by the server. In this case, whenever Alice and Bob collect vintage bits, at least one of them also takes a larger random sample of the beacon, at a rate of order 1 in 10^8–10^9. The beacon server also certifies (for example by public key signature or hash pre-image [1, 11]) a random sample of the broadcast taken at a similar rate, which it publishes following sufficient delay to guard against the possibility of a replay attack. The beacon can sample blocks randomly, rather than individual bits. Alice or Bob can now guard against a false beacon by verifying (say, more than 80% match) sufficiently many of the bits which by chance occur in both server and client samples over the course of the collection period.

The number of shared bits increases linearly with the size of the sample being collected. Sampling at a rate of 1 in 10^8 for a base transmission rate of 10Tbps will thus require the beacon to certify about 1Gbyte per day. (If Alice also samples at the rate of 1 in 10^8 then over 80 Merkle bits will be shared per day.) There would be no technical difficulty for the beacon to send this amount of data down the optical medium given the terabit rate of the system. The beacon sample should be broadcast along with a sufficiently long hash, which is signed for authentication. However there is no real-time restriction on the broadcast of the signed hash, which may take place offline. The clients need to know that the beacon was authentic only before they commit to using the newly collected shared key, which as we indicated above takes a few weeks to a few months. This time scale also makes it feasible to employ authentication based on physical security (e.g. the delivery of physically authenticated records on tamper-evident media to the clients' sites) as an alternative.

The trust assumptions in our fibre-optic approach are very limited, and are nearly the same as those of the competing quantum approach: the beacon has to be trusted to be authentically random, and a man-in-the middle attack must be detected by end-to-end use of a weak secret. However we make no assumptions about the physical integrity of the fibre-optic link.

While the idea of cryptographic use of a beacon is not in itself new, previous work has tended to focus upon satellite implementations. The threat model for the fibre optic context introduced here is rather different to that for the satellite, and the ramifications of this should lead to interesting new developments.

References

1. R. Anderson, F. Bergadano, B. Crispo, J-H. Lee, C. Manifavas, R. Needham, "A new family of authentication protocols," Operating Systems Review, 32(4):9-20, (October 1998).
2. S. M. Bellovin, M. Merritt, "Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks," Proceedings of the I.E.E.E. Symposium on Research in Security and Privacy, Oakland (May 1992).
3. Bruce Christianson, David Wheeler, "Merkle Puzzles Revisited – Finding Matching Elements between Lists," Security Protocols 9, LNCS 2467: 87-90 (2002)
4. Bruce Christianson, Michael Roe, David Wheeler, "Secure Sessions from Weak Secrets," Security Protocols 11, LNCS 3364: 190-205 (2004).
5. Xuhua Ding, Daniele Mazzocchi, Gene Tsudik, "Experimenting with Server-Aided Signatures," in Proceedings of Network and Distributed System Security Symposium (NDSS'2002).
6. N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, "Quantum cryptography," Rev. Mod. Phys. 74, 145-195 (2002).
7. S. C. Gobby, Z. L. Yuan, and A. J. Shields, "Quantum key distribution over 122km standard telecom fiber," Appl. Phys. Lett. 84, 3762-3764 (2004).
8. R. J. Hughes, G. L. Morgan, and C. G. Peterson, "Quantum key distribution over a 48 km optical fibre network," J. Mod. Phys. 47, 533-547 (2000).
9. U. Maurer, "Conditionally-perfect secrecy and a provably-secure randomized cipher," Journal of Cryptology 5:53-66 (1992).

10. U. Maurer and Cachin, "Unconditional Secrecy against Memory-Bounded Adversaries," Crypto '97
11. R. C. Merkle, "A digital signature based on a conventional encryption function," Crypto '87
12. C.J. Mitchell, "A storage complexity based analogue of Maurer key establishment using public channels," in C. Boyd ed., Cryptography and Coding - Proceedings 5th IMA Conference, Cirencester, December 1995, Springer-Verlag (LNCS 1025), Berlin (1995), pp. 84-93.
13. Michael Rabin and Yan Zong Ding, "Hyper-Encryption and Everlasting Security," in STACS 2002, Springer LNCS 2285.
14. B. B. Wu and E. E. Narimanov, "A method for secure communications over a public fiber-optical network," Opt. Express 14, 3738-3751 (2006)
15. A. Yoshizawa, R. Kaji, and H. Tsuchida, "10.5 km fiber-optic quantum key distribution at 1550 nm with a key rate of 45 kHz," Japanese J. Appl. Phys. 43, L735-L737 (2004).
16. Z. Yuan and A. Shields, "Continuous operation of a one-way quantum key distribution system over installed telecom fibre," Opt. Express 13, 660-665 (2005)

Vintage Bit Cryptography (Transcript of Discussion)

Alex Shafarenko

University of Hertfordshire

This may be a highly controversial talk, because this is an area where things are periodically rediscovered. But in the process of reinventing it, I think we've found a few interesting protocol issues, and a few interesting technological issues, which make it worth revisiting.

What's the idea of vintage bit cryptography? It's not an established term, it has other names, but the key idea is that a quantity of information too large to be stored anywhere on this planet is effectively an unusable secret. What's a secret? A secret is something that nobody has a copy of, or no bad guy has a copy of, right. Certainly nobody has a copy of this secret because it's too large to be copied, and it's unusable for the same reason. So the key principle is: public transmission of the unusable secret, with subsequent private selection of a small usable sub-secret. So secrecy comes from two things, the fact that you can't store the whole thing, and that you don't know which random selection of it has been taken.

James Heather: How can you transmit the thing if it's too big even to store it?

Reply: By generating it randomly. It's too big to store, but it's not too big to transmit over a significantly long period of time. The amount of information too large to be stored is technology dependent; this whole approach is technologically dependent. In a sci-fi world, if you have 10^{20} bits of storage on a PC, this approach doesn't work, at least at the transmission speeds that we have available at the moment. But the interesting thing is that the paper by Mitchell [1] goes back to 1995, and he gives some figures about the cost of storage, transmission rates, etc, and the figures that we have today still support the principle. Ten years in this area is a huge amount of time.

So here are some assumptions. I think the assumptions of this method are more important even than the method, because there are several variations. The first assumption is the availability of an open authenticated channel between all members of the key group. We're talking about key distribution: this protocol is used for agreeing a secret key within a group of people. So they need to have an authenticated channel, which doesn't have to be secret, confidential, it could be an open channel. The second assumption is that the public broadcaster has high entropy, so is genuinely random. If it doesn't have high entropy then it can be compromised easily by knowing the information basis of the broadcast from which the whole broadcast can be reconstructed. If you can compromise the authenticated channel, then the original approach collapses, so in that approach

[1] Referenced in the position paper.

you need to secure the authenticated channel very well. (We'll show later how to remove that assumption.)

So how is it done? The original method was to publicly agree a long observation period, then Alice and Bob collect a random selection of m bits each, and they don't tell anybody which bits these are as they collect them. Then they use the open authenticated channel to check which bits are common between them. Now if you do the sums then you will see that quite a lot of bits that they collect are common between them; if they collect m bits each out of the M that are broadcast then the number of bits that they have in common is about m^2/M. These bits form the secret key. They openly tell each other which positions they have in common because the large unusable secret is already gone, it can't be stored, and if you didn't know which bits to collect you wouldn't have collected them.

If there's an interceptor, Eve, then she also has her own random selection, say of n bits. These bits would have bits in common with both Alice and Bob, but the proportion of those would be small, assuming that M is such a huge number that you can't collect that many bits, so n would be much smaller than M. So Eve doesn't get very much; for practical purposes Eve gets nothing at all with a high probability. This is wonderful because no secret communication between Alice and Bob takes place, and yet they've agreed a common secret key, and nobody can intercept that, and the only thing that we need to secure is an open authenticated channel.

The first thing that comes to mind is that this is not a very scalable procedure because if you have lots of pairs of users exploiting the same public broadcast — and the public broadcast is an expensive thing, so you would want to share that — then what's the probability that an arbitrary user can intercept the private key of a pair that he doesn't belong to? We all know about the birthday attack, so this is not a small number necessarily, and the public source is expensive so we would like lots of people to share that.

Now with protocol issues, the original paper admits that if the open authenticated channel is there then you can use Diffie-Hellman and agree a private key, so what's the point of having a public broadcast? There is a point, still, because Diffie-Hellman may in the future be broken (quantum cryptography, all the rest of it), and the proposed scheme is free of all that, it gives you genuinely random keys. However, if you do have an open authenticated channel of good quality, you can assume at least that Diffie-Hellman will not be broken over the observation period, which in the original paper is one day. Also it's one day so you're not short of time; you can spend an hour of that day to run a really huge Diffie-Hellman, which is hard to break even if you have some trick up your sleeve. And the complexity of Diffie-Hellman is linear in the size of the key. So, maybe all you need to do is to have a common secret, a small shared secret between Alice and Bob, and use that for a common random selection, then you don't need to rely upon the collisions between selected bits, and so that makes it unnecessary to use any protocol subsequently, you just share a secret, and after an observation period you have a strong key which is common to Alice and Bob.

Another reason to make this assumption is that an open authenticated channel (in the absence of the trusted third party, public key infrastructure, and all the rest of it), does involve sharing some sort of secret, so a shared secret is already assumed in a way. If you don't want to use public key cryptography, why not use the shared secret for a common random selection.

But there's a much greater problem. In fact it is, strictly speaking, insuperable in the scheme that's being proposed. It is impossible to prove that the public broadcast has sufficient entropy. A bad guy can replace the public source by a random number generator, note the seed of the random generator, wait for the shared secret to be used to collect a strong key, and then use the random generator again to recover the strong key completely. And it is impossible for the broadcaster to prove that it is genuine, that it doesn't use any random, pseudo random generator, and it is possible to disprove by the user. So you need a technological solution. I can't see any kind of cryptographic solution here.

Another problem is the fact that high bit rate sources (and we need one for public broadcast here), are prone to errors. So Alice and Bob will not have exactly the same bits from the same random selection; they will have a large key with a high proportion of bits common, but some not: maybe up to 10% errors, if we use the technology that I will touch upon later. So you need some sort of protocol to agree a common key when you have *almost* coincident bit strings.

Well in fact it's not difficult. Here's your bit key $K = K1|K2|K3$ that you've collected from vintage bits. The part that you want to be the shared secret key is $K1$, then you sacrifice $K2$ and $K3$. Alice calculates a forward error correction check-sum $F(K1|K2)$ based on her value of $K1|K2$. That FEC check-sum would be such that it could correct a large number of errors — it has enough redundancy. We don't want to disclose any information about the secret key so Alice XORs the FEC with her value of $K3$. This may not be the same $K3$ as for Bob, because $K3$ also has errors. But fortunately the forward correcting algorithms do not require you to have a clean check-sum; a check-sum can also have errors, and that doesn't prevent their recovery. All you need to be worried about is that the strength of the error correcting algorithm is sufficient to recover from errors in both.

OK, so $A \longrightarrow B : FEC(K1_a|K2_a) \oplus K3_a$. If the strength of error correction is sufficient then B can compute A's value for K1, and this is the shared key. Now we need to verify that it's the same key, that the error-correction has worked. We can do this at the same time as removing the requirement for an authenticated channel by including a strong cryptographic check-sum, so $A \longrightarrow B : h(K1_a) \oplus K2_a$. If the first step worked then Bob has the same value as Alice for K2, so he can recover Alice's value for h(K1) and compare this with the hash of his own value for K1 which doesn't have to be the same for Alice and Bob. So at the end of this the protocol either fails, because there were too many errors, or it succeeds. Now the technology that we'll talk about a bit later is reliable enough that you can almost guarantee that it always succeeds.

OK, so what's the technology? I collaborate with one of the leading fibre-optic groups in the world, I think. At Aston University they've got a huge experimental

set-up worth millions, and they also do lots of theoretical research, and so I called them up three days ago and said that I needed something fast, and not necessarily reliable. They said, well that's a problem, we can give you 5 terabytes per second, but no better than 10^{-3} bit error rate. I said, I'm happy with 10^{-1} bit error rate. They said, well that may be 10 terabytes per second, but we won't go further than across the Atlantic. I said, I don't mean across the Atlantic, I mean locally, between the main headquarters of the bank, and the branch, maybe 200 kilometres. They said, well we haven't researched that, maybe 100 terabytes per second. They just split the optical range in the fibre in 2000 channels, and they send 5 gigabytes per second down each channel, and they're very worried about interference between different channels. If I want a random stream then I needn't be worried about that, because it can't get less random due to that.

Audience: I just want to query the idea that it can't get less random. If you have the sort of interference where one channel is completely wiping out another channel, then that would make it less random.

Reply: Ah, that's not what happens. What happens is that the bit error rate, which is normally 10^{-6}, becomes 10^{-2}. It's a non-linear medium, highly unpredictable, and there's also noise from amplifiers, from repeaters, from all sorts of things; there's a huge amount of technology sitting behind it.

So that's very interesting for them, nobody asked them before for a bad fibre which is fast, people usually ask them for a good fibre, and don't mind it being a bit slow.

The other problem that they find with optical fibres is that they're bursty, errors come in clusters. We absolutely don't care about that either, because we make a very rarefied selection, we take one bit out of 10^{17}.

So what's feasible storage? People from the Grid project can correct me, but last time I checked... there's this European collider project which requires the computational grid to store collision data from a year's worth of observation, and they reckon to need 10 Petabytes, and that's huge. OK, let's assume 100 Petabytes is unfeasible, just for the sake of the argument. Then the question is, how weak is the weak secret? If the weak secret is very weak, then I can just do the observation for each value of the secret, instead of observing the whole stream. Basically if you have N bits in the broadcast, and you need k bits of key, then assuming that you have a good random generator, you need $\log_2 N - \log_2 k$ bits of weak secret. Even under aggressive assumptions 60 bits would secure vintage-bit cryptography. What's 60 bits — a 10 character password. They can share a 10 character password, then after the observation period they can publish that password, right, because you can't go back in time and collect those bits that you needed to have collected.

George Danezis: So how could you actually generate good randomness of the public broadcast?

Reply: Oh, well, just use a resistor, heat it up, and trigger some digital device, you will get good randomness; I promise you can't repeat it.

Michael Roe: I know that type of device, and it turns out to be quite hard to get an acceptable number of random bits out of them because it amplifies all kinds of horrible things and you end up having a sort of driven oscillator.

Ross Anderson: That's a separate engineering problem about which much is known, but the way I see this as you present it, you're not actually doing broadcast randomness, you're really competing with the quantum crypto guys, who say, give us a fibre from London to Geneva, and we'll send you a key. That is what makes this different from Maurer.

Reply: Yes, yes, that's exactly right. I have a slide about that at the end, because there's more than one very interesting technology that fits the vintage bit sort of scenario.

We need a countermeasure for the low entropy problem. What can we do? We can use a reflector instead of a public source, so each guy in the key group generates their own randomness at a smaller rate, if you have a thousand users, then you just generate one thousandth of the public broadcast bandwidth.

Mike Bond: What do you mean by reflectors? Are they an abstraction?

Reply: No, it's a physical device. The reflector just combines streams from these sources.

Mike Bond: So it's a mirror?

Reply: No, it's a transponder essentially. It takes information from all these channels, which are different wavelengths on the same fibre, for instance, and then interleaves them. If it doesn't have enough sources it can interleave some random sources as well, of it's own doing. The more users you have for this system, the better it works. That's one half of it. The other half is a monitoring loop, each of the users computes mutual information between the bits that were sent to the reflector, and the bits that were received in the same positions. Suppose there's a random interleave, for instance, using a public formula, so we know where our bits are. The reflector can't guarantee that these bits will be intact, because there are users that will collide with those bits, or there may be also some random content that will collide with this, but we can measure the amount of mutual information, and if it drops below the critical level we can raise the red flag, after the observation period. The crucial thing here for safety of the cryptographic solution is that we don't communicate at all during the observation period, because we want to make full use of the huge shared secret. OK, now just note that neither the reflector, nor the channels to the reflector, are trusted. This doesn't matter; you can intercept everything, you can forge anything you want. Because of the monitoring of mutual information, you will be found out eventually. It's statistical of course, it's all probabilistic.

Now quantum key distribution has been mentioned, and Toshiba is selling a solution; it's no longer in the lab, it's a product. You have a piece of fibre, and a technology that guarantees that a single photon emitted by the source is received by the receiver without anybody intercepting it. If somebody intercepts it, this will be detected. The cost of this technology is tens of thousands of euros, depending on what you buy, per user. It has a reasonably high performance — it actually transmits about 100 secure bits per second, which for cryptographic

applications is quite a large number. Bruce drew my attention to the fact that it is actually quite prone to the man-in-the-middle attack, just cut the fibre, get the man-in-the-middle, you're done. That's why they need integrity and end-point authentication on the side-channel. You could continuously monitor the fibre, using all the other measures that you can take to prevent man-in-the-middle, but if you can do that then you don't need quantum key agreement in the first place. In fact we all know a power failure goes a long way in these schemes!

There's also talk about satellites. Why can't we have a satellite broadcaster up in the sky and use it as a source of vintage bits? In fact we can't. My first instinct was to use a TV broadcast satellite, it's got about a thousand TV channels, each channel around 2 Megabytes per second, so this is in the order of 10 Gigabytes per second. The problem is that all the contents of the broadcast is recorded somewhere, and if you need a selection you just go back in time and ask the content providers to give you a copy. However, there's new and exciting technology called UWB, you probably have heard about it, because it's going to be used for PCs, short-range communication, but it is particularly good for satellites.

We are coming to the period in technological development where the term "frequency" will fall out of use. Now the content will carry itself, there's no carrier. What will happen is, you will have pulses of electromagnetic energy, 200 pico seconds in length (that's a very short pulse), with a duty cycle about 64K, so there will be 64K slots, on average, between two pulses. The amount of energy that accumulates in one pulse is such that if you just consider these pulses, you could hear them at the end of the solar system. But because you hear noise in between, you can't hear them. So the same principle applies, the pulses are positioned according to some sort of random sequence, if you know that sequence you can hear. By the time you've cracked the password you've already missed all the bits, so it works on the same principle.

Now if you don't want a geostationary satellite being used as a source for reasons of entropy (because it could be compromised, an evil government can control it), then to get free of this problem you can have a swarm of micro-satellites on Low-Earth-Orbit. You have a satellite orbiting the earth not very far up, like 100 kilometres, and now the earth will rotate underneath. Now if you have, say, 40 satellites here, in the same orbit, then what happens is that you will always have a satellite overhead. The footprint of this scheme is about three or four hundred kilometres and within that area all the users will see the same satellites. Now each micro-satellite can be completely dumb, it could be mass produced, it doesn't even have to have any kind of intelligent electronics, or radiation protection, because all it needs to do is create digital noise — not analogue noise because that's hard to deal with, but digital noise. Now to compromise this scheme, you would have to compromise a significant number of satellites because users can XOR several of them. You can have not 40, but 400, in that orbit. They are the size of a grapefruit and the cost of launch is about 10,000 euros per kilogram, which is competing with the quantum distribution fibre-optic solution from Toshiba. The satellite itself will be free, though, because

it has no intelligent satellite guts in it. It's just a simple electronic circuit, which actually is faulty as well, because it's not radiation protected; but that's OK, you can't get more random than random. And I don't think this is compromisable by any kind of realistic means. In science fiction you just fly your spaceship to each of the satellites and replace it, but since NASA tracks all satellites the size of grapefruit and above, this will be known; if you just touch it, it changes orbit.

OK, conclusions. Over the last ten years, the storage to speed ratio has not changed much. We can still do vintage bit cryptography, and that encourages me to suggest that maybe we will be able to do it for the foreseeable future. Despite the fact that I've shown two schemes, I don't think satellites are a good solution because of the cost of management implications, and trust issues. However, I must say that a fibre-optic broadcaster is entirely possible, to the extent that we're going to construct one, with the guys from Aston University, and demonstrate it.

Bruce Christianson: We think we can undercut the quantum fibre optic product.

Ross Anderson: You don't need a broadcaster, you need a fibre-optic link?

Reply: Yes, a broadcaster in fibre-optic, so there will be one source and lots of receivers along the same fibre.

Mike Bond: Sorry, did you assert earlier on that your scheme is invulnerable to man-in-the-middle in the same way as the Toshiba scheme is, or that it isn't?

Reply: I did assert it given that we have a reflector on the source. The first experiments will be with the source, then we'll try and engineer a reflector.

Mike Bond: Using cable cutting between people and the reflector, why can't the attacker assemble everybody into a virtual subnet with 49 other imaginary people, and 50 reflectors? So everybody talks to their own reflector, and 49 other fake people?

Reply: The amount of mutual information that the key group receives from the reflector is known by calculation. We monitor what the reflector throws at us, and any random sample from any other people...

Mike Bond: But each person's monitoring their own?

Bruce Christianson: I understand what you're saying: the channel over which the protocol runs — you remember the XOR and the forward error-correcting scheme — that's not over fibre-optics, that's end-to-end coverage.

Ross Anderson: You need some authentication somewhere.

Bruce Christianson: Yes, you do, but so do the quantum people. You need to know that you're listening to the correct source. This typically involves a conventional side-channel.

Ross Anderson: Which brings us back to the problems of having the authentication end-to-end. We know the quantum crypto guys have different ways of doing this, by looking at hashes of sub-streams they receive and checking that the hashes are the same; presumably at least one type of bootstrap from password will do that. You can't do it many times from the same password though.

Bruce Christianson: The key point is that the quantum people have this same problem, and there we can use the same techniques that they're using. They have to have a weak shared secret for authentication. For the next authentication, we can use some of the new strong secret.

Reply: Actually the communication between Alice and Bob doesn't need to be authenticated at all, it only needs to be authenticated if you want to deal with denial of service, for no other reasons.

Bruce Christianson: Yes, that's true.

Michael Roe: Don't you also have to know that the secret that you end up with is shared with the person you think it's shared with, rather than with the attacker?

Ross Anderson: The authentication there is implicit, because the authentication is not going to work if you chose different sub-streams of the random source. What you're demonstrating here then is yet another way in which to parlay a weak shared secret into strong authentications, namely using your mechanism of very long bit streams, and transmitting sub-streams with their error correction bits.

Bruce Christianson: Unlike the quantum people, we get that for nothing.

Ross Anderson: OK, so you should bring out in the paper that you have got yet another alternative to EKE.

Bruce Christianson: Yes, that is a good point. I think the other point worth making is this: the first step is for users to get from the weak secret to a strong secret, and then to use the strong secret where they would have otherwise used the weak one to authenticate. That way they can do it as many times as they like.

Ross Anderson: Yes, OK. So perhaps what one ought to do is write this out formally as a paper, and point out that you've got an error correction-based protocol for authentication.

Bruce Christianson: Yes, that's a good way of putting it actually, because that makes the novelty clear.

Audience: A second question I had was about your storage requirement assumptions. When you said 100 petabytes, I remember doing sums for a look-up table for DES, which was about 500 petabytes for a single ciphertext. I was thinking, gosh, I wonder if the NSA has got 500 petabytes. My question is, given the special requirements of keeping the data just long enough to be able to look back and get the bits you want just in time, could there be any specialised storage, for instance, like the equivalent of mercury delay lines set at solar system scale, or could you send it all into space?

Bruce Christianson: Or use slow glass[2].

[2] Bob Shaw, Light of Other Days, Analog, August 1966. In the story the refractive index of slow glass is about 1.5×10^{19}, as light takes 10 years to travel a quarter of an inch. Lene Vestergaard Hau et al, Nature 397(1999) p 594 describe real slow glass with an RI of 3×10^{10}, about 120 feet per hour. However for real slow glass the product of delay with bandwith is fixed for a given cross section, so it is still worth investing in a picture window, rather than a single thin fibre.

Mike Bond: Just keep it spinning in optic-fibres for long enough.

Bruce Christianson: Or bounce it off a deep space probe, and send it back again.

Reply: The observation period is not necessarily limited to one day; the longer it is, the more insuperable the acquisition problem becomes.

Mike Bond: What's the regional range for satellites at the moment, presumably only a few light hours?

Bruce Christianson: There's some probes that have already left the solar system. And maybe aliens will reflect our broadcasts back at us[3].

Reply: You won't get my signal back. This is my first attendance at a security protocols workshop, and Bruce warned me to expect intellectual paranoia. But this is paranoia on a galactic scale. You need a 70 metre deep-space network dish to communicate with something that's that far.

Bruce Christianson: The key point is that you need to have already launched the probe some time ago.

Reply: Yes, the whole strength of this approach is that it is retrospective, all of it, yes.

Bruce Christianson: The advantage of this approach is that magic buttons invented tomorrow don't help the attacker against bits you have already laid down.

Reply: Exactly, you need a time-machine to break this; we could call it time-machine security.

Jolyon Clulow: I'm still not clear what the difference is between this and Michael Rabin's proposals for a fleet of satellites.

Bruce Christianson: Well, we didn't say much about the fibre-optic case in the talk[4], the short answer is that the threat model is different for fibre optic. But in all cases the trick that makes it work is that you can't store all the potential key material at once.

Ross Anderson: So what precisely are the security semantics of strings that are too long to store? We've seen now several examples of things that we can do with them. Suppose you have got an Oracle, which is privileged over normal mortals in that it has infinite memory, what special tricks can we make this Oracle do, what sort of new complexity tasks can you conjure up to keep the theoreticians busy for the next 30 years?

Reply: What a wonderful thought, that's a good question to end on.

---

[3] Probably starting with the BBC Third Programme.

[4] Largely because we hadn't yet filed the patent application.