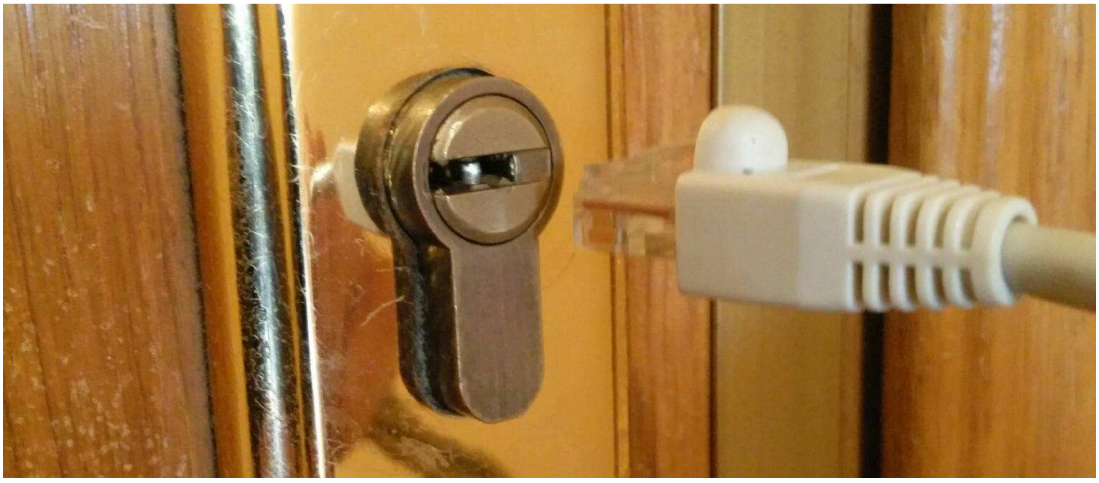
Internet security and privacy beyond 2016

adolfomaria.rosasgomez@gmail.com www.adolforosas.com

March 2016



We feel that we need 'security' in our networks especially in internet. But what do we mean by 'security'? Do we have clarity about what we want? There are concepts that live in the security arena but are not equivalent: reliability vs. trust vs. safety, identity vs. anonymity vs. privacy.

By increasing the scale, reach and speed of our networks we are redefining what we used to name as 'communication'. Let us explore the paradoxes that internet is introducing in human communication. Let us take a look at what we can/cannot expect today from internet and let us look beyond 2016 for the possible evolution of human communication through internet.

Modern day trade-offs

In today's News you can easily find 'Famous Government' vs. 'Famous Company' in a clear case of privacy vs. security, you can recall a recent breach that exposed data of customers of a business in which privacy was key in a case of privacy vs. morality, in recent years we have seen information leaks suggesting that governments perform massive scale interception of citizens communication, so the leak is a case of government right to keep secrets vs citizen right to keep secrets.

The Room

When thinking about communication in The Internet I propose to apply a simple paradigm: imagine that all the actors in communication are people in the same closed room and they have no other device than their voice and the ability to walk around. Anyone can speak aloud and publicly for all the room or N people can join together and 'try' to talk in short range volume. I propose this simplified model to analyse all possible cases of privacy, anonymity, and trust in today's internet. Due to the current capabilities of internet HW and SW and due to its wide reach there are many similarities in terms of privacy and trust between The Room and The Internet.

Reliability and Privacy

We want the network to deliver the message to the intended recipient and no one else, but worldwide networks like internet cannot grant that a single message will ever reach the intended recipient. This statement has a meaning beyond technology prowess. High quality fibre links have virtually no losses, connection oriented protocols like TCP are supposed to grant that every message makes it through the network, but routers, fibre links and TCP connections can be spied and attacked by anyone with physical access to them (anyone can spy in the case of internet).

VPNs and shared secrets

VPNs (Virtual Private Networks) are subnetworks created 'on top' or 'through' internet. They are kind of a private tunnel through a public space. Access to the VPN is restricted to N endpoints that share some common property. VPN technology adds reliability by making more difficult to break or spy the conversation. VPNs fight impersonation, tampering, injection or deletion of messages. VPNs rely on encryption (cyphering messages) but encryption is not completely safe under attack. There are many methods to cypher data today. All of these methods rely on pieces of information or 'keys' that must be known only by the two entities that communicate. Message plain text is combined with the key using some transformation that is difficult to invert without knowing the key. It is difficult but not impossible. Inverting the transformation must be straightforward if you know the message and the key. The most obvious cipher is based on a symmetric key. The same key is used to cipher and to decipher. Having key + plaintext the direct transformation is easy to do and it renders a ciphered message. Having the ciphered message and the key the inverse transformation is easy to do and it renders the plaintext. Symmetric key cryptography requires that sender and receiver have the key. There is a ´key distribution problem'. The transformation selected must be very difficult to invert when you have the ciphered message but not the key.

Statistical attack

As far as the same key is applied to a growing amount of information (many messages) the channel between sender and receiver becomes more and more vulnerable to statistic attack. In everyday´ s life keys are short pieces of information compared to the amount of information that these keys encrypt. As Claude Shannon demonstrated the only possible escape to statistical attack is to use a key that is at least the same length of the message it cyphers. Shannon´s demonstration led to an encryption technique known as 'one time pad'. Sender and receiver have a shared secret (key) as long as the message. Once the key is used to cypher a message, the key is discarded (and thus the 'one time '). To send a new message the sender must use a new key.

Everything is a VPN

Beyond TCP we could use any imaginable protocol to make our channel through internet more resistant to losses and less vulnerable to attacks and/or spies, but from a logical point of view any imaginable 'secure' channel built through internet is completely equivalent to a VPN and always relies on some kind of encryption, so it shares the vulnerability of known VPNs to statistic attack.

A VPN is only as robust as its encryption technique. Establishment of a VPN link is based on the existence of secrets shared by both ends. State of the art VPNs use short keys that are static. How do we share these secrets? If we use internet we can be spied and keys can be discovered.

Public keys

A proposed solution to key distribution is public key cryptography. This is the solution adopted in certificates and state of the art VPNs. I want to share a secret (key) with many people. I divide the key in two parts. I distribute part 1 widely (public) and keep part 2 secret (private). Anyone having part 1 can use it to cipher a message and send it to me. I can use part 2 to decipher what was ciphered using part 1, but no one having only part 1 can decipher it. If I want to reply to a message I need to know part 1 of the receiver´s key, his 'public' key and he will use part 2, his 'private' key to decipher. This is not really 'sharing a secret' as public keys are no secret, everyone knows them, and private keys are never shared. The relation public key-private key is what mimics sharing secrets. It mimics sharing because it exports some information about part2, the private key, without exporting the whole key. The methods used to divide a key in public + private are difficult to invert when you only have the public key and the message but do not have the private key, but inversion is not impossible, it is only computationally difficult.

Out Of Band secret sharing

An alternate approach to public key is a key distribution method based on 'out of band secrets'. This 'out of band' means that we need to share a secret (the key) with the other end by means of any other channel that is not internet. Two people in The Room can communicate aloud in front of everyone else with perfect secrecy as far as they have shared enough secrets out of The Room.

Grid Cards

As you can verify, people that need privacy over internet channels have put in place VPN-like mechanisms that rely on out of band secrets: Banks provide 'pads' (grid cards) to their customers, cards with an indexed sequence of characters. With each new transaction the bank provides an index and requests the customer to provide the indexed character. This mechanism uses the OOB secret to authenticate the user before establishing the channel. A grid card cannot hold many different keys, so the reservoir of keys is pretty small to implement an OTP (one time pad).

E-Tokens

Some companies provide e-Tokens. Every e-Token is a portable device with a reasonably accurate local clock that is one-time synced to a central server at 'device-to-user-linking-time'. Every e-Token generates a short pseudorandom sequence (PIN) based on the current time and a seed. Every e-Token uses the same PRNG algorithm to generate the PIN. This mechanism ensures that we can 'share' a secret (PIN) OOB (not using the internet) between all tokens and the server. The server ´knows´ how to generate the PIN based on the current time-slot and it has the same seed, so it can check if a PIN is good at any given time. When a user needs to start a secure VPN to the server the user can add the PIN to his identity to qualify as a member of the server + e-Tokens

closed group (a kind of 'family' of users). This authentication mechanism is called 2-factor authentication (password + PIN) or multi-factor authentication. This mechanism works as far as the PRNG algorithm remains unknown and the timestamp of the e-Token cannot be recreated by an attacker. The PIN is only valid and unique inside the current time slot; usually the server allows slots to be 10s to 30s long. Quartz IC clocks in e-Tokens have considerable drift, and they cannot be reset by user so if there is no resync at the server side for that user account (and there usually isn´t) after some time the PIN authentication will fail. To overcome this limitation a better quartz clock (more expensive) can be used or the server may try to adjust to the drift of each specific user by maintaining a drift number per user and adjusting it with each new PIN authentication request. As you can see it suffices to reveal the PRNG method and seed to compromise the whole network, as it is not really difficult to recreate a valid timestamp to feed the PRNG inside a 30s slot.

Connected e-Token

A refinement of the e-Token is the 'connected e-Token'. This is a portable device with a clock, a PRNG, memory and CPU with crypto functions and a communication link (more expensive). The physical format may be a smart card or it can even be an App living in a smart phone. The connection to the server solves the drift problem, and that is all the merit of the device. Crypto functions are used to implement cyphered protocols that handle the synchronization. These crypto functions will normally use a symmetric cypher applied to the PIN extracted from the PRNG. As you can see the connected device does not protect the ´family' (the set of users that share the server) against any attack that reveals the PRNG method. An interesting property of some connected e-Tokens is that they can be used to generate PINs in sequence, one per time slot, provide them over a USB link to a host and the host will use them to cypher a sequence of transactions (which is faster than entering the PINs by hand). The connected e-Token adds a weakness not present in the e-Token: synchronization takes place in-band, so it can be attacked statistically. Now there are two ways to attack the connected e-Token: 1) discover PRNG method, 2) spy synchronization messages. By means of 2) an attacker can solve 1).

Secure transaction vs secure channel

As you can see bank grid cards and e-Tokens just protect the start of a session. They protect a single transaction. The rest of the session in the VPN is protected by a static key. No matter how securely this key is stored, the key is short compared to the message. Connected e-Tokens may protect a small number of transactions per minute. Latency token-server limits the minimum size of the time slot in which a PIN is unique. So forget about apps that have more than 2 to 6 short messages per minute. In Internet physical access to the links cannot be avoided. This means that all the messages can be captured and analysed statistically. The current usage of bank pads and e-Tokens provides just an illusion of privacy to users. The best we can say about grid cards and e-Tokens is that the less they are used the more secure they are against statistical attacks. But hey, the most secure transaction is the one that never happened, so did we need to buy an OOB device to re-discover that? Definitely these devices will not work for people that want to ´talk a lot' privately through internet.

Identity and perfect Trust

We want to ensure that our message reaches the intended recipient and no other, but at the same time we know that there are many entities in the internet with the capacity and the motivation to detect and intercept our messages. (Remember The Room). Again the only perfect method to determine identity based on messages received over internet is 'shared secrets´. We need to ask the other end about some information that only this other end can know. As we have discussed above, OOB secret sharing is the only method that can grant perfect secrecy. Authentication (determination of identity) today can be done with perfect reliability as far as we have an OOB channel available (for instance we can walk to our banks desk to get a grid card or we can walk to our IT help desk to get an e-Token). Authentication is easily protected by a small shared secret because it is a 'small and infrequent transaction'. It carries little information and we do not do it 10^6 times a second, so it may be enough to replenish our shared secrets reservoir once a month, or once a year. The problem that comes with current implementations of perfect authentication via OOB shared secrets is that this method is 'only' used to secure 'the start' of a connection (a VPN or a TLS session), and it is never implemented as an OTP, because keys are reused: grid cards reuse keys as the card does not hold many keys, e-Tokens have a much wider key space so they reuse less, but knowing the seed and method you could reproduce the actual key at any moment, so the 'real' key is just the seed and that seed is reused in every transaction. To simplify let us assume that we implemented OOB secret reasonably to protect the start of the conversation, we 'started' talking to the right person, but after the start an attacker may eventually break our VPN by statistical attack, then he can read, eliminate or inject messages. The right solution would be to apply OOB authentication to every message. Clearly the grid card or the e-Token or the connected e-Token do not work for this purpose. Can you imagine hand-entering a 6 digit PIN for every 6 chars that you transmit? Can you imagine chatting with the other end at a pace of 6 chars every 30 s? It does not look very attractive.

Can we have perfect Trust? Trust usually means we can be assured that the message is not modified and identity of our partner is known. We cannot protect any internet channel of a decent bitrate using the available OOB secret sharing technology available today. So no, in general, we cannot have perfect Trust. For a reduced amount of transactions or for a low bitrate we can use one time pads. Two partners can agree to meet (OOB) physically once a year and share, let's say 1TB, 1PB, whatever size they require of secret data in a physical device (HD/flash memory), and then they will consume the secret over the next year having perfect secrecy. OK, that works. But as noted it is a fastidious technique and it has not been implemented mainstream.

Anonymity

Anonymity in communications may have two meanings: 1) I communicate to N receivers, no one of the N can know my identity, interceptors cannot know my identity; 2) I communicate to N receivers, all of them know my identity, interceptors cannot know my identity. As far as at any time during my communication I can explicitly reveal my identity the important difference between 1) and 2) is that 1) presents a mechanism in which a receiver must accept a message

from unidentified sender (as in telephony) while in 2)there cannot exist unidentified senders but there are identity hiding techniques oriented to interceptors. Internet today is in the case 2). It is not possible to hide the origin of a message. It can be tracked, always. There are mechanisms to obfuscate the identity of the sender (Tor network), but these methods only make the task difficult and this difficulty can be overcome using a decent amount of computational power.

Do we really want anonymity? Anti-tampering

In the phone world there is no real anonymity as any call can be tracked by the carriers if there is motivation (a court order for example). But out of those extreme cases, it is possible and really annoying to receive calls from unidentified callers. Many people have developed the tradition of not taking unidentified calls, which is a very reasonable choice. In internet it is not really possible to hide the sender address. Yes, there are people with the capability to do 'spoofing', tampering with the lower level headers and faking the sender address in a message. This spoofing technique looks scary at first sight, but then you must remember that the address of the sender, as any other flag, header or bit of information in a message is unprotected in internet and can be tampered with. That tampering capability means that the message can be modified, faked or even destroyed, but it does not mean that the message can be understood by the attacker. Without understanding the message semantics it is easy for the two ends that communicate to devise mechanisms that alert of tampering: checksums, timestamps, sequenced messages, identity challenges, and many others. These mechanisms will use OOB information so they cannot be attacked statistically. So, no, we do not want or need anonymity and we are not afraid of message tampering as far as we have enough OOB secrets and we know how to build an anti-tampering protocol based on them.

Current levels of Trust

It is interesting to note that the internet that we have in 2016 does not have what we demand from it in terms of security. As we have briefly reviewed everyone is in need of a VPN to connect to every other person or service. This is not happening yet. Even in case of a hypothetic VPN boom tomorrow morning, every commercial VPN is vulnerable to statistical attack, so we will be just reducing the set of attackers that can do harm to those with patience and big computers: governments?, big corporations?, organized crime? Can we really implement VPNs based on OTPs that in turn rely on OOB secrets? Well, we can do it on a one-by-one basis, so if we meet someone in the real world and we have periodic access to this person in the real world we can replenish our OOB secrets and conduct perfectly secret VPN traffic. But as you easily see we will not like to do that for every relationship that we have today through internet with everyone and with every service that we use. And by the way, current commercial VPNs do not implement proper OTP.

Devaluation of privacy, identity, responsibility and trust

So no, in internet we don't trust. We can't. Those with private info, important transactions, or a lot of responsibility know how to use OTP based on OOB secrets. Those who don't, maybe you, probably are not aware of the solution or the perils of not having a solution. The result is people

do not expose through internet those bits of information that are really valuable to them, unless they have no other option. If you suspect that your bank's grid card is not secure enough for your operations you have very little option beyond doing every transaction personally at your bank's desk. To buy a book via internet you are not going to worry. If you are target of an online fraud you will take it as a risk of modern life. If someone impersonates you on LinkedIn or Facebook, things may get more serious. You may end up in court. Even in that case what can you do? Are you going to ask LinkedIn or Facebook to implement OTPs? I don´t think so. How could they do it? Will they have a LinkedIn or Facebook desk in every small village of the world to share OOB secrets with 100 billion users? We are seeing increased usage of VPNs. Especially for remote workers. We are also seeing increased usage of multi-factor authentication, naturally for VPNs and remote workers but that is also becoming common for wide spectrum services like Facebook, Gmail, LinkedIn, and others. Trust is 'forced'. We trust online retail platforms because we want to shop online. We cannot live without that. But we will not shop in the first online portal that we bump into. Prestige in the online world is more important than ever. Companies that have been longer in the market and have a track record of none or very little leaks of data or compromised transactions will be the choice.

What to expect in the near future

Internet evolution is accelerating. Many companies that are in business today will not be in 5 years. Many companies that do not exist while I write this will become dominant in 5 to 10 years from now. In terms of security we cannot expect a breakthrough in such a short time. We may see some sophistication reaching the common internet user. We can expect to have free personal VPN services with state of the art security, which is not really 100% secure, but it is what 'free' will buy you in the short term. VPNs for businesses will grow in security. The best of them will opt for higher levels of encryption, maybe even OTP/OOB. Services that have a wide range of users will target multifactor security for authentication and transactions. They will surpass soon the current level of security that we can find in banks.

Banks, they need to evolve.

Banks really do not seem to be taking the challenge very seriously. The technology that they use is way old and insecure to be dealing with customer's money. As the non-banking world evolves providing electronic payment we can assume that banks will improve their technology to attract customers. One of the first movements must be to provide VPNs to all their customers and better OOB, more complex secrets given at their desks to consumers. Grid cards are no good for protecting frequent transactions. As micropayments for online services become much more popular (they are already popular now), and thus much more frequent, grid cards need to be replaced by an OOB method with a much wider key space. I do not think e-Tokens are a good replacement. Much better would be a gigantic grid card implemented as an indexed ROM created per user. Let´s say 32-64 GByte of random bytes burned on an inalterable IC given to every customer. Add a display and keyboard to enter the index and you are done. This kind of IC can be

created today and is affordable to banks. The eGridCard must not have connectivity. Any connectivity will make it weaker as the keys could be spied over USB, or wifi or any kind of link.

Social and Retail

Multi-factor authentication will take over the place. Social networks do not have a great return on each individual member (a few cents/year due to ads), so they are unlikely to invest in a HW OOB+OTP, but I can see cheaper multifactor coming: adding phone numbers to your identity (the phone network is a good OOB separate channel). I also see premium services coming from social networks. Paid premium services allow to provide OOB+OTP HW, as described for the case of banks. Online retail sites and online premium social networks can offer true privacy to their members via eGridCards, at least to protect start of session. To protect long messages we will need a better way to share a huge secret.

Professional big secret sharing

Corporations wanting to seriously protect a channel, not a transaction, will push the state of the art of VPNs. Combining VPN methods for reliable channels: sequencing, timestamping, identity challenges, checksums, multiple handshake, and others with OOB+OTP will make corporations much safer. This effort will require new HW and new SW. In opposition to protecting a single transaction, protecting a channel requires a continuous feed of secrets to the transmission device. This feed cannot be delegated to a human (as in an e-token or Grid card), but we cannot rely on an 'open interface' as USB, Ethernet, radio or whatever existing link. The solution that comes to mind is that the secret holding HW must be coupled to the internet connected device only briefly, while the channel is active, and the coupling must be a one way interface that does not work from the internet side. This kind of HW interface is not available today (at least it is not mainstream) , but there is no difficulty in building it.

Size of secrets

We can speculate that any 'decent' communication today is very likely to move from KBytes per minute to MBytes per minute. Non-media-intensive talks will be in the lower 1 Kbps to 100 kbps, while state of the art media-mixed talks may be 100Kbps to 500 Kbps, and some rich-media talks will reach the Mbps (1 Mbps to 5 Mbps). This speculation applies to very general communication taking place in social media, micro transactions in banking and retail (small photographs included), in mobile terminals and desktop computers. In other more professional environments like VoIP and videoconferencing we may move up the Mbps scale. If we want to protect a channel of 1 Mbps that is active 8 h/day, 300 day/year, we need 8.64x10^12 bits (8.64 Tbits =1.08 TBytes). It will be easy to build OOB shared secrets worth of 1 TByte/year. A cheap HD will do.

Internet fabric

Internet is made of routers and links. We have said that every link and every router is accessible to eavesdroppers today, which is true and you better act as if you believe in that statement. Internet is multitenant (many entities own the routers and the links) so we could reasonably guess that

some internet portion could be hardened against eavesdroppers while remaining standards compliant in its connection to the rest of internet. Yes, this can be done by replacing every router in that area with routing machines that cipher every single packet that goes through using OOB + OTP secrets. Ciphering works end to end in the area that the secret is shared. As this area cannot be the whole internet, we can think of a new kind of router that admits periodic replacement of a HW storage module containing OOB secrets. All routers in the area will receive the module let's say once a week or once a month. Modules will be written centrally at network owners premises. Traffic that comes into that 'spot' in internet will be ciphered via OOB+OTP so only routers in that 'spot' will understand the low level packets. Egressing traffic will be just 'normal' traffic as low level packets will be deciphered at the border. The 'spot' will be transparent to the rest of internet, but now traffic cannot be spied in that spot. This is a business advantage. If a customer traffic originates in that spot and terminates in that spot it will be much more secure and the customer does not need to do anything special. This claim may attract final customers to an specific ISP or Telco or network service provider. This could be called a STN (Secure Transaction Network) for similarity to a CDN, which is a closed infrastructure. Today we call SDN a Software Defined Network. Interestingly SW defined networking will make much easier to build custom routers and thus STN. Imagine how easy it will be to build a 'device' out of a fast packet forwarding engine (HW based) plus SDN modules for OOB+OTP written in house to cipher every packet and support our proprietary storage module. I would move from my current ISP to another ISP that can swear (and demonstrate) that my traffic will ONLY go through this kind of routers in my country. At least I can reach my bank and a good number of friends and family in a secure spot.

Standards

It is very unlikely that we will see a new standard appear to include ciphering in the base internet protocols to transform all routers in secure routers. Even if we see that standard appear in the next few years (5 years) that standard will be based on classical cryptography which is vulnerable to statistical attack. This is due to the impossibility of specifying OOB mechanisms in a standard. And due to the fact that very few global coverage networks exist that are not internet accessible (OOB). The most practical two networks that can be used for OOB are: people carrying secrets in their pockets, phone (non-data but voice) network. The second network is much less reliable as an OOB than the first one. Even if an agreement is reached for a OOB method (impossible in my view) adoption through a significant part of internet will take over 10 years, which will render the effort useless.

Conclusion

You have to do your part. If you want to have an increased level of privacy you cannot count on current privacy protection from internet links and/or routers, internet protocols, bank grid cards, e-Tokens, or VPNs. You cannot count on this situation improving to a practical level over the next 5 to 10 years. You can implement some sort of OOB + OTP today on your own. Just look for the pieces of technology out there to implement your secret sharing at the level that you require.