Usable security:

Managing both technology and humans

Thomas Quillinan

let's change
YOU. US. THE WORLD.

THE HAGUE
UNIVERSITY OF
APPLIED SCIENCES

Usable Security

The Centre of Expertise in Cyber Security[1] is a multi-disciplinary team of researchers that are working together to develop expertise, insights and know-how in the field of Cyber Security. The Centre of Expertise is made up of three lectoraats: Cyber Security and Safety, lead by Marcel Spruit; Cyber Security in Small and Medium Enterprises, lead by Rutger Leukfeldt, and finally Network and Cyber Security, lead by the author.

The lectoraat in Network and Cyber Security is actively researching three main areas: Identity and Access Management; that is, how can we develop with novel approaches to both identify people and systems, but also control access to them. This is a fascinating topic for me, and it was also the topic of my Ph.D. research[2]. I have always believed that if you can properly manage the access a subject has to an object, you can fundamentally secure that object. I am interested in both the technological and societal aspect of this control: that is what technologies can be build and how can we best manage people so that these technologies will correctly operate - more on this later.

Secondly, security for the internet of things. I am particularly interested in how we can use these devices towards developing new applications - and how to manage the security and privacy of the data coming from these objects. We have already some very interesting results in this area, our preliminary work has focused on how much data is being produced by these devices. I will note that I'm not talking about securing the devices themselves - this is both the easiest aspect of IoT security and the most difficult as we are talking about devices that are impossible to update and are already compromised. Instead, I am most interested in the use of these devices, both in the most frequently cited area: smart homes, as well as the more industrial use cases including Smart Cities, Smart Energy and Industry 4.0.

Finally, secure usability; my interest in usability comes from the fact that security is seen as a non-functional aspect of systems. That is, it is seen as an add on "just sprinkle some security dust over the application and everything will be fine". Surprisingly, that doesn't often work. Another area that suffers the same fate is Human Factors, the study of how to design systems to properly account for how people will use them. Adding to this challenge is the fact that these

1 https://www.thehagueuniversity.com/research/centre-of-expertise/about-centre-of-expertise-for-cyber-security

2 T.B. Quillinan, Secure Naming for Distributed Computing using the Condensed Graph Model, July 2006, UCC, Cork, Ireland.

two areas are often seen as in conflict. It is common to think of securing a system will result in making it harder to use.

I argue then that "Usable Security" is the study of how we can best balance the needs of security with how the users of that system wish to use it. Usable Security is the merger of Human Computer Interaction (HCI) and Security. My view is that if you don't do this, it will result in a system that is both insecure and unusable. To butcher a famous quote from Benjamin Franklin:

> **"Those who would give up essential usability to purchase a little temporary security deserve neither usability nor security."**

For the remainder of this document, I want to first examine the problems I see with the computer security world, with respect to how systems are built and then give you a view into how I believe we can address these challenges.

Contents

The Problem with Computer Security

In general, the problem with computer security was, in my view, best expressed by Professor Ed Felten of Princeton University:

> **“Given a choice between dancing pigs and security, users will pick dancing pigs every time.”**

That seems like an outrageous statement, but came from an experiment that Prof. Felten ran in the early days of the Internet. He created a Java applet showing a pair of dancing pigs. As can be seen in Figure 1 below, it wasn't very exciting. This applet is a simple program that runs inside your browser. However, before it could be run, it displayed a warning: “The Applet “Dancing Pigs” is requesting additional privileges. Granting the following is high risk: Modifying files stored in your computer”. Prof. Felten then measured how many people clicked “OK” versus how many people started the application originally. The answer, as suggested by his quote: 100%.

Figure 1 *The Dancing Pigs Applet.*

Bruce Schnieier, CTO of IBM Resilient, and a prominent Computer Security Researcher went further. He stated that:

> **"The applet DANCING PIGS could contain malicious code that might do permanent damage to your computer, steal your life's savings, and impair your ability to have children,"**
>
> **A user will click OK without even reading it. Thirty seconds later they won't even remember that the warning screen even existed.** [3]

This is a tragedy. Furthermore, typically someone like me, a trained computer security researcher, will tell you that it's the fault of the user. People who know me know that one of my favourite statements in this regard is that I have a solution to all of the worlds security problems, that I have borrowed from Bender in the show Futurama. It's very simple: "Kill all Humans". Problem solved. If we are all gone, then the systems we designed will work perfectly, forever.

Seriously though, I do not believe that it is really the fault of users - the system architects and engineers have to shoulder some blame. Another quote I like (from American Author Rich Cook):

> **Programming today is a race between software engineers striving to build bigger and better idiot-proof programs, and the Universe trying to produce bigger and better idiots. So far, the Universe is winning.** [4]

3 Bruce Schneier: Secrets and Lies (John Wiley & Sons, 2000; ISBN 0-471-45380-3), p262

4 Rick Cook, The Wizardry Compiled (Baen Books, 1989; ISBN 978-0671698034)

Einstein went further - he said:

> **“There are only two things that are infinite: The Universe and human stupidity, and I’m not yet completely sure about the Universe.”**

You can surmise that I believe that users are idiots. However, this is an oversimplification. The reality is that each of us is an idiot some of the time. Being an idiot has nothing to do with intelligence - it has more to do with the lack of clarity about the design and use of a system.

To make the situation worse, the easiest thing in the security world is to say no. Security experts are trained to minimise risk as much as possible, and therefore saying no becomes the easiest solution to most requests. Furthermore, we are typically among the most paranoid people in the room. Therefore, when we are asked to look at a system, we see all the dangers, and are not in the same frame of mind as the end users. This is the major flaw in system design: users are not typically part of process. Sometimes this is necessary, Henry Ford once said that

> **“If I had asked people what they wanted, they would have said faster horses.”**

Steve Jobs said that:

> **“Users don’t know what they want.”**

Computer Security Principles

Even from a security experts point of view, there are fundamental problems. As security engineers, we are taught to look at the world through a number of fundamental models. For example, the CIA triangle - nothing to do with the spy agency, but the concepts of Confidentiality, Integrity and Availability. Confidentiality refers to the need to keep information secret unauthorised actors; Integrity means ensuring that information is not tampered with, and Availability is the requirement that information is accessible to the authorised users.

I've mentioned authorised users a couple of times - that adds a fourth topic to the triangle! Authorisation, or Access control, as I mentioned earlierv is an area of particular interest to me. Access control [5] is concerned with providing control over security critical actions that take place in a system. Providing control over actions consists of explicitly determining either the actions that are permitted by the system, or explicitly determining the actions that are not permitted by the system.

There is one further 'A' that is important: Auditability, the requirement that we know when, why, and how something happened. The most common attributes are shown in Figure 2.
This leads to yet another topic: non-repudiation, the study of how to prevent someone claiming they didn't perform an action. There is also the consideration of Privacy.
What I'm trying to say here is that we have a lot of tools and models at our disposal. The important thing is how do we deploy them to best address the security problems. We have all these fundamental tools and models. It's easy then. We just apply model 'A' to problem 'X' and everything is fine. Well not so fast.

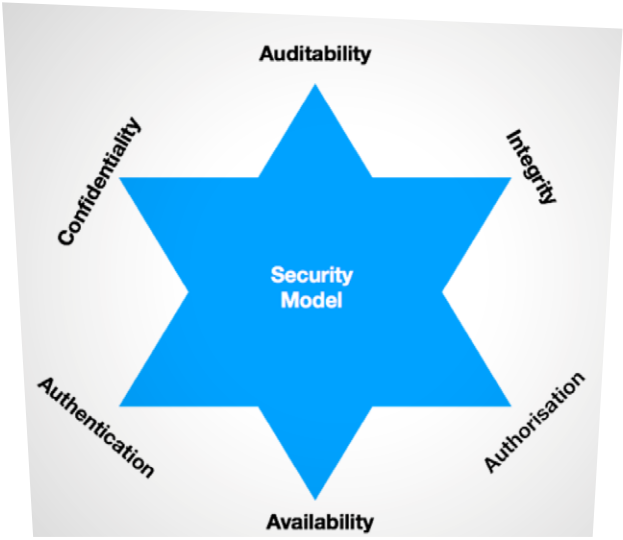

Figure 2 *The Most Common Security Attributes.*

5 D. Gollmann. Computer Security. Wiley, 1st edition, 1999. ISBN: 0-471-97844-2.

Authentication

The most common approach to user authentication is the standard username & password. A recent web comic on the site XKCD made

> **“Through twenty years of effort, we’ve successfully trained everyone to use passwords that are hard for humans to remember, but easy for computers to guess.”**

Password requirements are typically expressed as “you need to have at least one capital, one special character and one number in your password” [6]. The comic itself is shown in Figure 3. Worse still, often there are requirements that you have to change your password periodically, say every three months. What is the outcome: first people add their year of birth to the end of their password. People use common substitutions such as ‘0’ for ‘o’, ‘3’ for ‘E’, ‘@’ for ‘A’, and so on. These are very predictable and therefore easy to break. The people who came up with this advice originally, the US National Institute of Standards and Technology, have recently recanted it completely and now advise different approaches such as two-factor authentication.

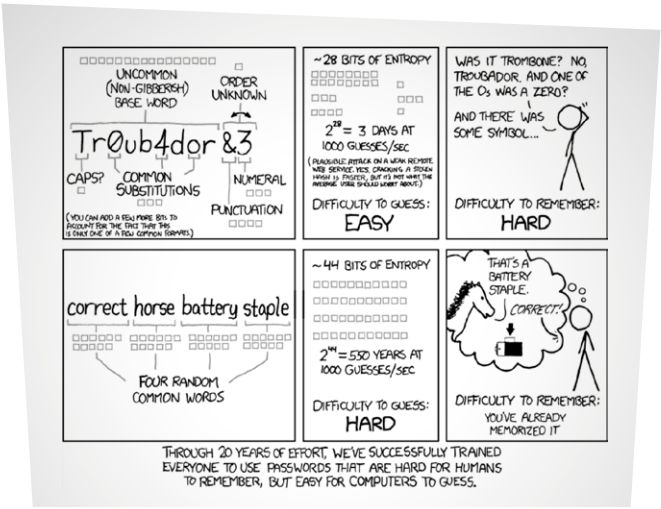

Figure 3 *XKCD #938, Password Strength.*

6 https://www.xkcd.com/936/

You often hear about two-factor authentication - the idea that we should use two different factors when authenticating ourselves to a system. These factors are normally broken into three areas: something you know - for example a password; something you have - for example a physical key, and something you are - for example your fingerprint. Common advice is to use any two of these factors to authenticate yourself to a system. I tend to agree with this, but with the proviso that you must be careful about the biometric side - that is something you are. It's very easy to change a password, people start objecting when we insist upon cutting off your fingers to revoke access!

A common type of two factor authentication is one of several types of phone application. These applications, such as Google's Authenticator [7], generate unique time-sensitive passwords that are linked to your phone and a website. Another type is the Dutch Government's DigID App[8] that scans a QR code on your screen and communicate with the backend to validate your login. Such systems provide excellent security but are not used by a large number of people. For example, Dropbox provide the ability to use the Google Authenticator. According to Dropbox's Patrick Heim, "less than one percent of the Dropbox user base is taking advantage of the company's two-factor authentication feature". [9]

Figure 4 *A Common Result of Difficult Password*

7 https://www.google.com/landing/2step/

8 https://www.digid.nl/en/about-digid/digid-app/

9 http://krebsonsecurity.com/2016/06/dropbox-smeared-in-week-of-megabreaches/

Another type of biometric token, is a picture of your retina. This is imaged differently to an iris scanner - where a simple picture of your isis is taken and compared. With a retinal scan, you need to shine a bright light into the back of your eye, and then take a picture.

One of my favourite stories that I've been told - and to be clear, I'm not sure how true this is - relates to physical security at a US base in Germany. The background: there is a communication room on the base that needs to be secured. We call these "red rooms", where secret information is stored and used. One of the requirements of such rooms is access control. You can solve this requirement in a number of ways: put a guard at the door with a list of approved people; lock the door and hand out keys, or other suitable tokens. At the base, they went with a retinal scanner. They are expensive and supposable harder to manipulate than iris scanners. There was only a single flaw: if you are going in and out of the room, each time you need to have this bright light shined into your eye. After a few times a day, this quickly leads to migraines. Fantastic security, poor usability! The problem was, naturally, solved. Someone got a doorstop and kept the door open all day. The first person to arrive would take the hit and then block the door. Fantastic usability, terrible security.

Security as a Tradeoff

We should also talk about when good design decisions are made. Security is always a tradeoff. Often times it is a tradeoff between perceptual security and reality. For example, consider the physical security at events. Often times people feel secure just because security is visible, without any real reason. I attended an event at the Stade de France in Paris last year - a France vs Ireland Rugby match. It took place not long after the Bataclan attacks in Paris, and the authorities had stepped up security. It took us nearly an hour and a half to enter the stadium due to the multiple pat down searches. We were "more secure" right? Well, in my view no - instead of people being in the stadium, separated out, people were crammed together outside. Rather than removing the threat, you simply move it to another location. This is what Bruce Schneier refers to as "Security Theatre".

Figure 5 *Immigration Control in JFK Airport in New York.*

Another Example is shown in Figure 5, the immigration control in New York's JFK airport. The obvious desire is to monitor all lanes, but there is a basic flaw, other than the ridiculous number of cameras shown here: you have to staff the security room with people whose job it is to monitor these cameras. This is an intensely boring job and has been shown that people are not

capable of monitoring boring systems over time. A simple example can be seen in an Awareness test that you can find on YouTube. [10]

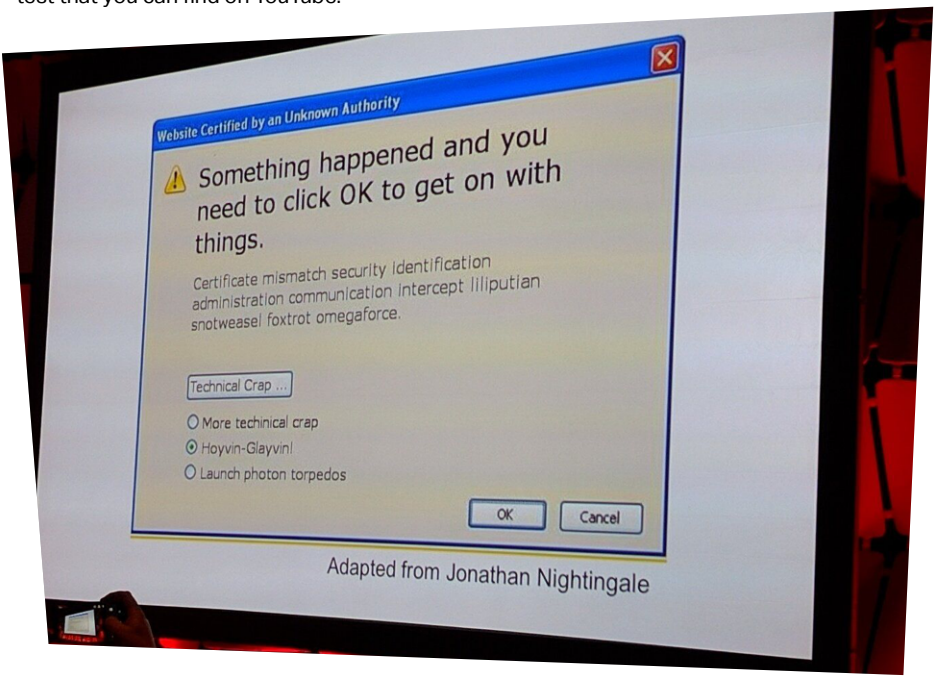

Figure 6 *How users understand warning messages.*

There many equivalents in Computer Security. For example, if you ask most people what the most visible security they see on the web, they will immediately mention the padlock symbol in the web browser. The interesting thing is when you ask them what it means. Most people do not know. How can we possibly make good security decisions when we do not understand what's going on? When users get warnings, they don't understand them, and therefore don't read them. Figure 6 is, while a joke, an example of how users read these messages. The basic response is, I don't understand this message, so just click 'OK' to get on with my life.

For the record, the padlock means that there is an encrypted link between your web browser and the web site. Secondly, it means that the web browser manufacturer trusts the certification authority, the organisation who verified the owner of the website. You will note, there is no mention of the user in this process. You are implicitly trusting three parties here: the website owner to properly implement their website; the browser manufacturer to only trust good certification authorities, and the certification authority to properly manage their security.

10Awareness test: https://www.youtube.com/watch?v=47LCLoidJh4

On the first, we all know that this is unreasonable - unless you have a dedicated security team, it is very difficult to keep up to date with the security flaws that are discovered daily. As for the browser manufacturers, they do not have any real reason to make dramatic decisions about which authorities to approve or not. I counted the number of authorities in a default set of a well-known browser: there were 85 unique authorities trusted. This included a number of government authorities, including the Netherlands! What this means is, if the Dutch government wanted to, it could create a certificate for a fake website posing as a legitimate one, and your browser would display the lock symbol. Figure 7 shows a comic with an unfortunate truth - most often, security incidents are headline news at the time, and shortly afterwards the company renames itself and continues as before. In DigiNotar's case, this did not happen. They went out of business shortly after the event.

Figure 7 *Fokke & Sukke Comic regarding DigiNoter, Translated, it says "Don't worry people, in three months we will have a new name and a new look".*

This can also happen by accident or by malicious third parties - indeed it happened many years ago to one of the biggest CAs, VeriSign. They generated a certificate for Microsoft. Unfortunately, they didn't properly verify the requester, who was not Microsoft. I was speaking to someone who worked in Microsoft at the time, and it cost somewhere around five million dollars to rectify the issue.

Finally, as for the certification authorities themselves, they have not been paragons of good practice either. A quick question, how many people here have heard of DigiNotar? Well, for those that don't, they are an infamous former certification authority, who happened to be Dutch. There are certain best practices for certification authorities - for example, having a good procedure for validating identities, ensuring that proper audit logs are kept, not keeping the secret signing key on an Internet accessible computer. DigiNotar unfortunately did not do most of these things. The good news: the browser and operating system manufacturers removed them from their list of trusted authorities once this was discovered. The bad news: there is no way to ensure that there is no other DigiNotars out there. Therefore, the lock symbol can be seen as security theatre too. However, I still think that it is better than nothing. Provided that people understand what it means and actually take the time to examine who they are trusting with their credit card.

Worse again is someone has recently pointed out that there is a character code for a padlock. You can therefore theoretically register a website domain starting with a padlock and, for example a legitimate bank's name and to the average user, it will look legitimate. We have spent too long telling people to look for the padlock without telling them why.

The final major challenge in computer security today is more abstract, and not something I will really address today. In any case, we need to have good laws and prevent knee-jerk reactions to events. I mentioned the physical security reaction to an event above. In the computer world, there are increasing calls, yet again, to restrict cryptography, to force manufacturers to create back doors that will, of course, only be used by the "good guys". Essentially, this can be summarised as: we need to ban mathematics for the bad guys. For some reason, I'm not convinced that will work!

To summarise, in my view, the biggest challenges with computer security today are threefold: helping the users make good decisions; helping system designers build intuitively secure systems, and helping the authorities to make good security policies. The common thread of these challenges is education - we need to properly educate the users, the engineers, and the policy makers. From my perspective, educating the users and the engineers can be best addressed by bringing them together. I will address this next.

Towards Usable Security

The issues described above have led to a dangerous aspect of system architecting and design: we know best, the users are idiots. Earlier, I mentioned the famous Steve Jobs quote how users don't know what they want. He also said:

> **"It's really hard to design products by focus groups. A lot of times, people don't know what they want until you show it to them."**

This is a subtly different quote, yet it results in a vitally important answer: you need to show users products. I don't agree with him that you can't design products collaboratively, you can but not by just presenting it to random people - you need to focus on the real expert users and design the solution with them. This is very difficult when you are designing for the general public, but more tractable when you have a specific community in mind.

This is hard. It means that you need multi-disciplinary teams to work on the problem. This includes technologists of course, but also psychologists, industrial designers, cultural anthropologists, lawyers and policy makers. I have been very fortunate while working at Thales to have worked with people from some of these disciplines, and one of the enormous benefits of working at a University is having access to even more. However, I want to first address how we can build such usable systems.

User-centric Design

In my view, the most important aspect of system design is to get "buy-in" from the end users. This means presenting prototypes of the system to the users in a series of workshops, and over time refining them towards the real system. This is known as "user-centric design" in the Human Factors world. Security is a difficult aspect of this approach, as users do not necessarily see the value of restrictions of functionality.

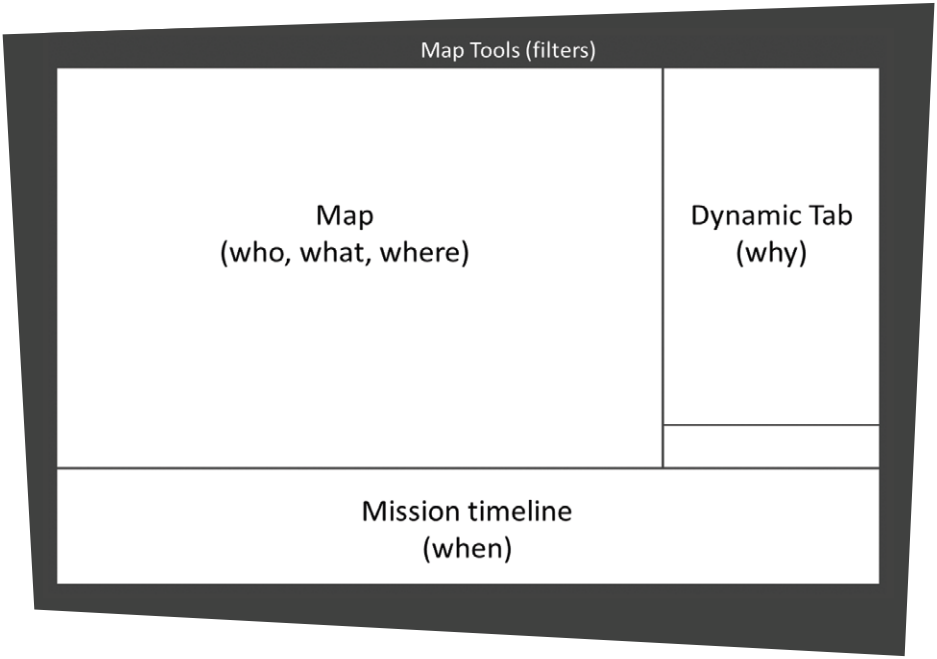

Figure 8 *An artistic Sketch of an Information Sharing System*

However, there is a class of users who do appreciate the need for security: the security officers themselves, including people like system administrators. If, for example, you are building a system that will require users to have different roles, this will be a vital aspect. User-centric design entails keeping the users at the centre of your design philosophy. It centres on understanding the operational needs of end-users and the way we could support their needs with technology. Security can be embedded into this process.

Typically, when we approach a problem in a user-centric manner, we arrange workshops with user groups periodically, to understand their operational context and to propose prototypes to them. The first workshop typically takes place at the earliest possible moment in the project.

Some artistic sketches are proposed, along with user stories that we use to motivate the examples. An example can be seen in Figure 8 [11]. Users are encouraged to explain their typical workflow, and how this could be reflected in the new system. The subsequent workshops consist of taking this feedback and updating the prototypes to reflect the updated user stories. Thus, a user-centric design approach is typically iterative in nature.

We can approach the security aspects at the same time. Firstly, looking at the authentication issue, we would attempt to identify the specific requirements of the users. For example, do they need to audit every action, and if so, why exactly; that is, do they need to have full traceability of actions, or what subset is needed. We can also identify how users should identify themselves to systems.

A good example of this is a research project currently being developed by The Hague Security Delta consortium, called the International Zone project. This project, a collaboration with Siemens, TNO and Thales, is investigating how we can help increase the collaboration between international organisations that The Hague hosts and the local police, in terms of the information that they share between themselves, and the host country. This is not a simple problem to solve, as it involves different jurisdictions and privacy concerns.

11 Courtesy M. Varkevisser, Thales Research & Technology, Delft.

Figure 9 *An RFID-based YubiKey*

One concern that came up during the security workshop was how can we authenticate all information entered into the system. We proposed a standard username/password solution. However, the users pointed out that the systems were shared between different shifts, and it was very unlikely that they would log out and then in again during shift changes. Furthermore, they didn't like the idea of administrating yet another authentication system. The final requirement was that all information entered into the system should be linked to a person, with only the organisation that entering the information having access to the identity of the person. This is a set of contradictory requirements!

While this was a setback in terms of design, we felt that it was vital to properly address the issue, and we investigated a number of solutions, such as requiring a PIN number when entering information. However, we came back to a simpler solution that would always work. We integrated a hardware token into the system, called a Yubikey[12] - a simple device that you plug into the computer and press a button to transmit a unique code to the system. The YubiKey shown in Figure 9 goes further and acts as an RFID reader that is used with the person's existing badge to generate the unique code. This resulted in a useable, auditable security solution.

Recently, I was also involved in a Cyber Security Summer School[13] here in The Hague, as an assignment host for a group of students. I asked them to look at the same problem, but from a completely independent viewpoint. They did a great job in the limited time allocated, and proposed a number of very nice alternative solutions, from using a set of unique devices that each person carries, called pico, to hardware tokens and biometrics. The takeaway is that regardless of the seeming impossibility of the task, we can come up with creative solutions.

12 https://www.yubico.com

13 http://summerschoolcybersecurity.org

The General Data Protection Regulation, GDPR.

There are some other solutions that are also worthy of attention. For example one of the major user requirements is the issue of the privacy of their data processed by a system. This is becoming increasingly important in Europe as the General Data Protection Regulations, GDPR[14], will be enforced starting on the 25th of May next year. The GDPR is intended to give EU residents more control over how their personal data is processed, stored and used by businesses.

From a business point of view, the GDPR is a huge deal. Fines for breaching the regulation can be up to €20 million or up to 4% of worldwide turnover, whichever is greater. For example, for Google potential fines would be up to $3.6 Billion; for Apple they would be up to $8.6 Billion. Not insignificant sums. Therefore, there is an increasing demand for both technical and administrative responses to the GDPR. Compliance and data protection officers are in huge demand, for example.

User demand for privacy is growing, and especially in Europe regulators are taking note and driving this change. It is therefore very important from research in both security and usability to investigate how to ensure both user privacy and data security.

On the technological side, the GDPR refers to pseudonymisation - a process that transforms personal data so that you cannot figure out the original subject without additional information. Typically, this is managed by creative use of cryptography. One method that I find particularly interesting is the idea of privacy enhancing technologies such as how Apple manage searches and speech. For example, when plotting a route on a map, they break the search into smaller chunks and do not store who each chunk is for. They use the same process for managing speech recognition.

14 https://www.eugdpr.org

Content Based Security

Another related technique that we have been researching for the last number of years is the idea of information-bound security, also known as content-based security[15]. The idea behind content based security is that instead of taking all of your data and putting it in a single location, behind a very high wall, we instead take each individual data item and protect it individually. By protection, we mean encryption: that is, each item is encrypted individually. Therefore, the challenge goes from protecting access to the database, to controlling access to the decryption keys.

One nice aspect about a content based security approach is that we can build distributed systems where we can easily share information without a single central point of control. This is vital as collaborative systems are increasingly important. For example, if you consider the requirements of small and medium enterprises, called MKB's here in the Netherland. They often do not have dedicated security professionals; they cannot develop new security rules for their IT systems. An information sharing platform where communities of interest pool their resources to address shared risks can be easily imagined.

Information sharing between organisations is important when considering these collaborative systems, but one important aspect is decentralised control. That is, when collaborating, no one organisation is the ultimate arbitrator of who can join or leave the group. People and organisations want to be in control of their own data, who they share it with, for how long and to be able at any moment to change their mind. Content Based Security solutions can offer the means to ensure the confidentiality of the data, but unless you correctly control access to the decryption keys, it is just another technical solution in search of a problem.

15 S.M. Iacob, T.B. Quillnan, and J.B. van Veelen, EP2890084: A Data Securing System and Method, 01 July 2015.

Decentralised Trust Management

This brings me back to Identity and Access control. One type of technology that particularly fits in this instance is the idea of decentralised trust management. Trust Management is an approach to constructing and interpreting the trust relationships among public keys that are used to mediate security critical actions. Credentials are used to specify delegation of authority among public keys, and are used to determine whether a signed request complies with a local authorisation policy.

Blaze et al.[16] defined trust management as "a unified approach to specifying and interpreting security policies, credentials, and relationships that allow direct authorization of security-critical actions" Trust Management is basically designed to answer the question "Do I trust principal X to do action Y?" A trust management system enables permissions to be associated with cryptographic keys. These permissions can be delegated by one key to another. Trust management systems must be able to navigate these delegation chains, linking a request to the authority required to perform the requested action.The idea behind trust management is that access rights are stored in cryptographically signed certificates and distributed to the users, rather than managed in a central computer. This means that when an access control decision is required, the user must then present the certificates that prove that they are allowed to perform the action requested. This sounds complicated, but it is can constructed to be easy to automate and distribute.

There are a number of advantages to this approach. First, it allows each organisation or person to independently design their own security policy. Data owners can therefore decide who, what and for how long to grant access to their own data. Secondly, it allows controlled delegated - that is, as rights are written in certificates, you can give some or all of your rights to someone else by writing and signing a certificate for them. This can of course be disallowed by the data owner. A third advantage is that you can easily see how the access right was granted: the system can track the path from person to person to show how it was granted.

Finally, such approaches allow unique policies to be defined. For example, I mentioned multi-party computation earlier. One related aspect is the idea of separation of concerns - that is multiple parties must act in concert in order to perform an action. Such policies can be easily codified in trust management certificates.

16 M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized trust management. In Proceedings of the Symposium on Security and Privacy. IEEE Computer Society Press, 1996.

With all of these advantages together, you can quickly see how they can be used to support decentralised data sharing, where each organisation delegates rights to people both within their organisation and in other organisations. Due to the delegation chains, this is easier to manage as cross organisational delegations can be made to the heads of the foreign organisation, and you let those organisations delegate the rights within their own organisational tree.

The Internet of Things

One item from my research agenda that I have not linked in is the Internet of Things. However, I view IoT as an extremely distributed computational system that does not have a central point of control, or unfortunately many security controls. My major interest is the data that comes from these connected devices. I wonder how much consideration of the GDPR requirements has been performed - in my view, these devices produce some of the most personal information about European residents possible. I can both see the need and the advantages of integrating some of the aspects of content based security, trust management, and multi-party computational techniques towards building new and secure applications, using the diverse types of IoT sensors and actuators available today.

The advantage to this approach is the possibilities that we can build systems using devices from many manufacturers for the users rather than the big companies. This will empower users to collaborate and delegate access to their own devices to others - but only when they want to. I can envision applications where groups form in localities, whether they are smart homes or smart industries to combine resources as necessary. It is vital that users trust their systems -they must have confidentiality, privacy, authentication, authorisation and integrity built in by design. Furthermore, the users themselves must be in control - I cannot see how central authorities will give both the flexibility and the confidence to the public.

It is also vital to consider the variety of disciplines that are needed to support such applications. First, of course, is both the technical hardware and software support that is needed. Secondly, you need support for both the usability and security aspects. We need to make the security easy to configure and "on" by default, otherwise it will be disabled. The Human Factor to security configuration is impossible to ignore. Personally, I have a firewall at home that I have never managed to fully understand!

Finally, we need support from some other aspects - one of the most important is the legal aspect. I mentioned the GDPR a number of times - correctly interpreting the possibilities and requirements is vital towards building usable distributed systems for the future. This is even more important when considering the fact that the envisioned users of such systems are not necessarily either technologically sophisticated or, like myself, have a good understanding of the legal aspects of what is proposed. I hope that the interactions here in the Haagse Hogeschool will re-enforce this aspect.

Conclusion

As I mentioned earlier, I believe that the only way to make progress in security is to use a multi-disciplinary approach to system design. We need to remove the problem from the user and concentrate on making the system both secure by default and easy to use. Returning to the password example from earlier, I believe that the concept of passwords themselves are flawed from a user point of view. Users should not be required to remember a different password for each site and system; although they should use different credentials for each. A more usable replacement is to use generated passwords that are stored in a password manager. It's a security tradeoff, as you now have one place with all your passwords, but that is not the major danger. Unless your threat model is state sponsored attackers, brute force is not a reasonable risk. Instead using passwords that are susceptible to easy guessing - using the common replacements is a far greater problem.

I am more concerned with both education of users and education of system designers and architects. With this in mind, we need the support of a variety of expertise, from hardware specialists, through data scientists to Human Factors, interface designers and finally to experts on cultural aspects and legal issues. Speaking about cultural aspects, it is amazing how western focused computer security. Being Irish, one that jumps to mind are the small differences, such as the words "mná" and "fír", which if you see on the door of a toilet you would be forgiven for thinking mná means male, and fír, female. A tip if you're visiting Ireland, it's the other way around! Cultural aspects in security are extremely difficult to predict and reflect once again why it is vital that users participate in the development of the system as a whole.

I hope you can now see how the three aspects of my research platform: identity and access management; Internet of Things security and Usability fit together and reenforce each other. As a technical researcher, I see my role as helping to adapt security solutions to use cases. One of those use cases is indeed the Internet of Things, but the work carried out by the Centre of Expertise in Cybersecurity related to multi-organisational interactions, such as within the Hospitals, as well as the waterschappen, are also primary targets. I would also like to mention our new lector Rudger Luekfeld, who we are fortunate to count as our colleague working in security of SME/MKBs. I look forward to continuing working with both Rudger and of course Marcel Spruit, in the future.

To conclude, my main message is that while security is not an add-on and it should be core to all systems: Usable security is the main concern. Ignoring the users and enforcing your policies will lead to frustration and then flouting of the rules. Working with users to design systems that are both secure and usable is the only way where we can hope to solve this problem. Furthermore, there are several technological solutions available, if we can combine them in smart ways, that can at least help us on the road to a more secure and usable future, without having to kill everyone! This is not a trivial challenge but one where industry and universities can work together to solve.

The Hague University

thehagueuniversity.com

internationaloffice@hhs.nl

+31 (0)70 445 00 00

Johanna Westerdijkplein 75
2051 EN The Hague, The Netherlands