UK Financial Institutions Stand to Lose Billions in GDPR Fines: How can They Mitigate This?

Paul Tobin
Computing Science
Dublin Institute of Technology
Dublin, Ireland
Email: paul.tobin@dit.ie

M. McKeever
Computing Science
Dublin Institute of Technology
Dublin, Ireland
Email: paul.tobin@dit.ie

J. Blackledge
Computing Science
Dublin Institute of Technology
Dublin, Ireland
Email: paul.tobin@dit.ie

Mark Whittington
Business School
University of Aberdeen
Aberdeen, UK
Email: mark.whittington@abdn.ac.uk

Bob Duncan
Business School
University of Aberdeen
Aberdeen, UK
Email: bobduncan@abdn.ac.uk

Abstract

The new EU General Data Protection Regulation comes into effect on 25 May 2018. The vast majority of financial institutions in the UK are woefully under-prepared to comply with this legislation. Current estimates suggest that UK banks could potentially suffer fines in the first year alone of over €5 Billion. We argue how a simple encryption mechanism, in conjunction with the use of an immutable database, can provide an unbreakable system strong enough to resist attack whether using in-house systems or cloud applications, which are notoriously difficult to secure properly. The client can personalise security locally, where this encoder is in addition to any public security provided by the Cloud service. Adopting this strategy will make break-ins, side-channel attacks and any security issues such as backdoors in public encryption algorithms, redundant. Our proposed system uses the "one time pad", which with modern technology can resolve opposition to one-time pad shortcomings from the 1960s. When this is used in conjunction with the immutable database, a full audit trail can be maintained, as well as a strong forensic footprint, both of which are often lacking where cloud is deployed. We will show how storage applications do not generate key distribution problems, a major problem normally associated with one-time pad encryption and normally cited as the main objection to this encryption paradigm. With this system total control is returned to the end user and should overcome most security problems.

Index Terms

Immutable database; audit trail; one-time pad encryption; GDPR; key distribution.

1. Introduction

In this work, we consider the impact of the forthcoming EU General Data Protection Regulation (GDPR) on financial institutions. This regulation comes into effect on 25 May 2018, and affects all companies who trade in any way with the EU, whether substantially or only in part. The reporting requirements in the event of a security breach are significantly more rigorous than anything we have seen before, in that a breach must be reported within 72 hours of occurrence. The reporting calls for the breached company to know which records have been accessed, which have been modified, and which have been deleted.

Some 5 years ago, the global average reported time between breach and discovery was 6 months! Today, the average stands at 146 days. In the case of companies who have been breached, over 85 percent of the breaches have been notified by third parties. This does not bode well for compliance with the GDPR. Failure to notify within 72 hours is enough to cause a company to fail to comply with the regulation, leading to exposure to potentially punitive levels of fine. For a simple failure to report on time, this can be as much as €10 million or 2 percent of Global turnover. For more serious breaches, the fines can double. And another breach tomorrow, means yet another fine.

Financial institutions, being highly lucrative potential sources of income are prime targets for attack. The vast majority of companies in the UK are thought to be inadequately prepared for this new EU measure, and financial institutions are no exception. Current estimates suggest that UK banks could potentially suffer fines in the first year alone of over 5 Billion Euros. Achieving a high level of security is a difficult challenge for any company, and where cloud is in use, the challenge is significantly more difficult to meet.

It is well known that once an attacker breaches a system, their primary goal is to eradicate or obfuscate the trail of their ingress by tampering with or deleting both audit trail and forensic data. Some attackers have the skills to do this without trace, but many others are much less skilled, and the resultant impact of their blunt actions, including the loss of audit trail and

forensic data, can have a catastrophic impact on the company, both regarding satisfying statutory audit requirements as well as business continuity.

Audit trail and forensic data are targets for all systems, but usually, in traditional distributed systems, it is often easier to spot than in Cloud systems, where instances are often spooled up and down with great frequency. Incorrect configuration can cause both audit trail and forensic data to be very easily lost, without the possibility of recovery. Where the instance is incorrectly configured, accessing and abstracting forensic data from a virtual hard drive can be all but impossible once an instance is shut down after use.

This new regulation is likely to expose many companies, and in particular, financial institutions to an increased level of risk. Between internal whistleblowers and the propensity for attackers to boast of their prowess, it is unlikely that many companies will be able to "Get away without a breach being noticed". Thus we must consider how we might effectively tackle this potentially serious problem. In Section II, we address how we might utilise an immutable database to address these security weaknesses. In Section III, we consider the merits of using cryptographic means to further strengthen the approach. In Section IV, we show how a modern one-to cloud one time pad application might be used. In Section V, we consider why the resulting framework might prove extremely difficult or even impossible to breach, and In Section VI we discuss our conclusions.

2. The Power of the Immutable Database

In any business, there are many areas of activity which need diligent checking and verification by an objective external person or organization, but in financial institutions, the need for audit is much stronger. Financial institutions present an extremely attractive target to attackers, who are always very keen to gain access to the liquidity offered by these organizations. Audits such as the audit of financial systems and results are mandatory. Where a financial institution uses cloud as part of their operating architecture, this presents new audit challenges. Cloud computing audit is a new, immature field and it would be surprising if there were not lessons to learn from the experiences — and failures — of audit processes and practices that have been honed over decades if not centuries [Duncan and Whittington, 2014].

Finding people with the appropriate skillset whenever any new technical area emerges is always a challenge — because for people to have both technical knowledge of the area coupled with competency in carrying out an audit is rare. Audit companies may seek to extend their audit competence into new technical areas, not just cloud audit, but perhaps environmental audit as another example. With over a century of experience in the development of audit tools and practices, these next need to be applied to a new technical domain. Another option would be for computing specialists to pick up an audit skillset. The logical outcome would be for audit firms to recruit computer cloud experts and seek to harmonise their skills with those of audit already embedded in the firm, although this might lead to a culture clash between accountants and cloud experts.

A long standing tool used by accountants is the audit trail and this phrase is already in the cloud computing literature courtesy of the National Institute of Standards and Technology (NIST) [Guttman and Roback, 2011], for example. It is worth noting that the same phrase may not carry the same meaning in both settings. Quoting from the Oxford English Dictionary (OED) [OED, 2016]: "(a) Accounting: a means of verifying the detailed transactions underlying any item in an accounting record; (b) Computing: a record of the computing processes that have been applied to a particular set of source data, showing each stage of processing and allowing the original data to be reconstituted; a record of the transactions to which a database or a file has been subjected". This subtle difference of definition is recognized by the OED. Accountants are members of professional bodies (some national, some global) with membership limited to those who have passed exams and achieved sufficient breadth and length of experience, such that they are deemed worthy to represent the profession. Audit is a key feature of these exam syllabi and the tracing back to the source of each accounting activity (the trail) is a foundational aspect of audit.

Whilst NIST provide a clear explanation of an audit trail in a computing security setting [Guttman and Roback, 2011], and in keeping with the OED definition (b), the understanding of the term in cloud audit research is much less precise and consistent. For example, [Bernstein et al., 2009], see the audit trail including: events, logs, and the analysis thereof, whilst [Chaula, 2006], provides a more extensive and detailed list: raw data, analysis notes, preliminary development and analysis information, processes notes, and so on. Indeed, [Pearson and Benameur, 2010], accepts that attaining consistent, meaningful audit trails in cloud is merely a goal rather than reality. More worryingly [Ko et al., 2011], points out that it is frequently the case that an audit trail will be deleted along with a cloud instance, meaning no record remains to trace back, understand and hold users to account for their actions and [Soares et al., 2014], provide useful details of the requirements for accountability. Indeed, the EU Article 29 Working Party [EU, 2012], suggest poor audit trail processes remain as one of the security issues not properly covered by existing principles.

It is all very well to establish an audit trail, but it is vital that an audit trail must have its contents protected from any adjustment. As [Anderson, 2008], points out, system administrators must have no power to modify it. Even with well trained and ethical individuals, this is not only good practice, because it is always possible that a hacker might be able to gain access, followed by gaining administrator status. Thus, the audit trail must be protected by utilising an immutable database (i.e., one that only records new activities but never allows adjustment of previous ones).

Duncan and Whittington [Duncan and Whittington, 2016], in previous work warned that failure to recognise the importance of the audit trail could lead to serious problems, and in [Duncan and Whittington, 2017], they proposed a solution to this problem specifically for cloud by utilising an immutable database to support the provision of an independent audit trail, to which they suggested that all system logs should also be added. While the concept was proposed to specifically resolve the cloud audit trail and forensic trail problem, this proposed system would also work well with other system architectures, both old and new. This approach satisfies the long understood accounting requirements of an audit trail, namely that no user, not even system administrators should be able to gain access to the audit trail. Thus there is a requirement to utilise an immutable database (i.e., one that only records new activities but never allows adjustment of previous ones).

This approach satisfies the primary goal for the successful development of a system to preserve both the audit trail and system logs. Next, by relying on the principle of separation of duties, this immutable database is placed into a completely separate system, which removes the danger of a leveraged attack once a system has been compromised.

This will, of course, make the immutable database itself a prime target for attack. However, the machine on which the immutable database runs should have no external web services or other means of direct access. There should be very high security on this machine, including an intrusion detection system (IDS), and it should also run a permanent monitoring system to instantly detect anomalies, providing instant alerts when such activities are detected.

But we already produce an audit trail in our accounting system, we hear you say. Why would we require another one? There is a perfectly good explanation for that. First, we have already highlighted the issues with retaining an audit trail in cloud, particularly where instances are freely spooled up, and down, to follow demand. Few people make proper provision to retain the full audit and forensic trail in these cases, meaning the full trail is frequently lost when the instances are shut down. Secondly, we recommend the original audit trail and forensic trails are retained. Attackers expect to be able to find the audit trail and the forensic trail, so that they can erase their tracks, however poorly they might carry this out. Absence of these will signal to any potential attacker that someone has already been in and they will quickly depart. However, where the attacker thinks they are in control, they will generally remain for much longer periods, leaving greater amounts of forensic evidence behind.

This leads us to the next Section where we will consider the addition of a one-time pad encryption process, based on earlier work [Tobin et al., 2017b], which will attempt to catch the point of entry of the attacker.

3. One-Time Pad Encryption

In 2000, the Regulation of Investigatory Powers (RIP) Act was introduced in the UK to limit and regulate the powers of the public bodies to carry out investigation and surveillance, including communications interception. The Act was updated in 2015 and outlined the limits of government agencies to collect and use data from the public. However, there are no proposals by the government to ban local encryption and companies will not be asked to insert a "backdoor" into encrypted data. With this mind, the following points are important:

- Many well-known encryption schemes that adhere to the Kerckhoff - Shannon principle - the enemy knows your system - are much weaker than publicly acknowledged [Kerckhoffs, 1883], and,
- Many German WWII Enigma encoding devices were available for sale after the war by the authorities and created what we call the *Enigma* syndrome - the UK authorities knew the encryption algorithms and hoped other countries would purchase them (Russia politely declined)!

If we cannot trust standard encrypting protocols, clients will have to encode data locally using a secure encryption system such as the one-time pad (OTP) encoder which protected conversations between heads of state during WWII. Provided it is used correctly, and the OTP is truly random, it can be shown that the encoded data cannot be broken.

3.1. Historical use of the OTP

A patent was granted in 1917 for a one-time pad generator created by Joseph Mauborgne and Gilbert Vernam [Bellovin, 2016]. Dr Steven M. Bellovin, a professor of computer science at the Columbia University School of Engineering, discovered in "Telegraphic Code to Insure Privacy and Secrecy in the Transmission of Telegrams", written by a banker, Frank Miller 32 years earlier, how a OTP could encode telegrams [Bellovin, 2011]. There is no written documentation to show if Miller actually implemented the OTP encoder in hardware.

Figure 1 (a) is a picture of the SIGSALY encryption system developed in Bell Labs by A. B. Clarke and Alan Turing during WWII to secure conversations between Churchill and Roosevelt [Bennett, 1983]. However, SIGSALY weighed 55-tonnes, generated a OTP type noise key and stored on vinyl. The record containing the key was then flown across the Atlantic giving rise to the key distribution problem. It was a small price paid for a system said to be unbreakable and encoded files from this system remain "uncracked" to this day. Another historical example of OTP usage was the "hotline", popularised in films as a red telephone but was in fact, a one-time tape (OTT) teletype machine used by Kennedy and Khrushchev during the

Cuban missile crisis. Figure 1 (b) outlines the main signal processing in our random binary number OTP generator and shows two chaos sources initialised by noise from a detuned data receiver integrated circuit. Initialisation is just a process which "kickstarts" the oscillators into action from a random voltage level, thus making it impossible to replicate the process.

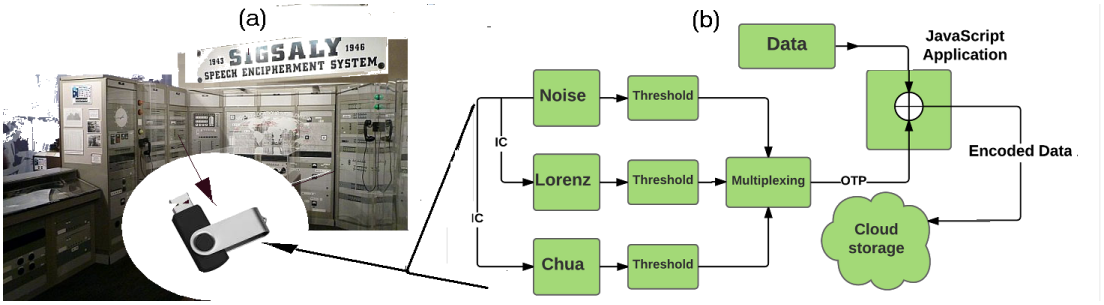

Fig. 1: Comparing the 55-tonne SIGSALY WWII encoding system with the OTP encoder using chaos sources.

3.2. OTP production

The random binary OTP sequences were generated using analogue chaos circuits housed in an external unit similar in size to a flash drive. Claude Shannon in his 1949 paper [Shannon, 1949] discussed how the properties of chaos signals could be used in cryptography to encode data [Blackledge and Ptitsyn, 2010]. However, in this discussion, we minimised the technical content, but the reader may consult [Tobin et al., 2016], [Tobin et al., 2017b] for details on how the OTP was generated electronically. A truly random binary signal stream consisting of ones and zeroes, is notoriously difficult to create and typically involves thresholding noise from natural sources such as cosmic background radiation, or lightning strikes. Thresholding is just an electronic process of converting the analogue signals to digital binary form. An example of this noise can be observed as specks on a detuned TV screen, or the hissing sound from a detuned radio. The proposed OTP encoder uses noise from a detuned data receiver integrated circuit to initialise two electronic analogue chaotic oscillator sources. The random chaos analogue signals when correctly thresholded, produced the desired random OTP binary signals [Tobin et al., 2016]. Receiver noise is not used as a standalone random source because it could contain non-random signals which a hacker could use to decode the encoded data.

4. Modern One-to Cloud OTP applications

Historical OTP systems discussed previously were one-to-one communication systems that had a key distribution problem. This eventually consigned the OTP to history and was replaced by symmetric block ciphers and asymmetrical public key algorithms [Rijmenants, 2009]. However, the speed and small size of modern electronic system means that a key distribution does not present any significant difficulties in modern cryptography, especially for one-to-Cloud applications, where there are no key distribution problems [Borowski, 2016]. The client physically uses the key for encoding and decoding data stored Cloud data at different locations. The key is transported using the Sneakernet method (This is a slang term when the key data is transported physically), i.e., couriered rather than sent digitally.

Figure 2 shows how to encode a document in a one-to Cloud application. This process is as follows: The end-user inserts the OTP encoder system into the computer and generates a OTP whose length is automatically set to the length of the data. The OTP should always be stored externally in an air-gapped computer (one not connected to the internet), or in a flash drive until needed. The OTP and plaintext data are then exclusively OR-gated and processed in a JavaScript located in the main computer. The Cloud Security Alliance (CSA) recommends companies should add authentication to the encrypted data to guard against breaches. Hence, a hash message authentication code (HMAC), such as SH-256, is added for verifying data integrity and authentication [Chandrakar, 2014]. Further software processing adds a von Neumann corrector to increase the overall entropy of the encoded data before storing in the Cloud. Decoding the downloaded data at another location is a simple process and uses the transported OTP to recover the data.

The following are typical applications where the OTP may be applied:

- Uploading exam scripts by academic staff for retrieval by office staff.
- Accountants are making a presentation about sensitive financial data in another country. Here, the accountant encodes his data using the hardware encryptor at home/office and then uploads it to the Cloud. The OTP is then carried by the accountant to decode data from the Cloud locally.
- Medical and legal one-to-Cloud applications are two further examples discussed in [Tobin et al., 2016], [Tobin et al., 2017b], [Tobin et al., 2017a].

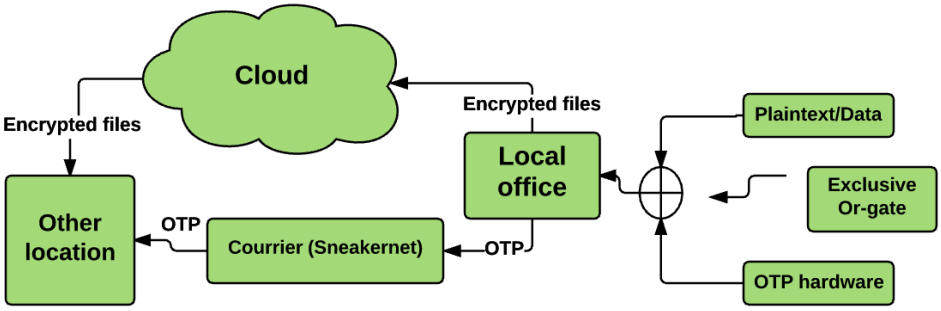

Fig. 2: Encoding data in One-to Cloud OTP application.

One-to-cloud applications have the following advantages:

- Download documents/images safely at other locations,
- Eliminates the need for transporting sensitive unencoded documents which could be lost in transit,
- Encrypting documents locally with a OTP prevents an intruder from understanding any intercepted encrypted data, and
- Avoid the punitive GDPR fines for late breach discovery.

5. Why the Two Approaches Combined are More Robust

The result of combining these two systems provides us with the assurance of a permanent audit trail, and a complete forensic trail being available to assist with the after effects of a cyber breach. This will allow financial institutions to adopt cloud use to obtain the cost benefits offered without the worry of the unresolved issues surrounding loss of the audit trail and forensic data. Whether using cloud systems or not, the level of protection offered to the audit trail and the forensic data means that any such system would be far more robust and resilient to attack.

However, it is certainly the case, that in the event of a breach, the financial institution concerned will have not just instant awareness of the intrusion, but will also be able to instantly identify which records have been viewed, copied, modified or deleted, one of the fundamental requirements of the GDPR, thus allowing them to mitigate the fine levied. Successful financial institutions tend to be large, and one of the consequences of a breach is usually that the loss of data can be massive.

6. Conclusions

We can see that ensuring a proper audit trail and forensic trail are properly maintained are difficult challenges, especially where cloud is used. It is clear that the vast majority of companies are likely to struggle to comply with the GDPR, yet be taking some relatively simple steps, it may be possible to reduce the impact resulting from a breach, thus mitigating the level of fines being levied.

We believe the state of readiness for the majority of companies, particularly those using cloud systems, is dreadfully poor. Considering that the majority of companies are unlikely to be able to even tell they have been breached for 146 days, they will likely be in breach from the moment the GDPR takes effect in May next year. This means they are effectively placing themselves in the firing line for a punitive run of fines.

References

[Anderson, 2008] Anderson, R. J. (2008). *Security Engineering: A Guide to Building Dependable Distributed Systems*, volume 50. Wiley.

[Bellovin, 2011] Bellovin, S. M. (2011). Frank Miller: Inventor of the One-Time Pad. *Cryptologia*, 35(3):203–222.

[Bellovin, 2016] Bellovin, S. M. (2016). Vernam, mauborgne, and friedman: The one-time pad and the index of coincidence. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9100:40–66.

[Bennett, 1983] Bennett, W. R. (1983). Secret Telephony as a Historical Example of Spread-Spectrum Communication. *IEEE Transactions on Communications*, 31(1):98–104.

[Bernstein et al., 2009] Bernstein, D., Ludvigson, E., Sankar, K., Diamond, S., and Morrow, M. (2009). Blueprint for the Intercloud – Protocols and Formats for Cloud Computing Interoperability. In *Internet Web Appl. Serv. 2009. ICIW'09. Fourth Int. Conf.*, pages 328—-336.

[Blackledge and Ptitsyn, 2010] Blackledge, J. and Ptitsyn, N. (2010). Encryption using Deterministic Chaos Encryption using Deterministic Chaos.

[Borowski, 2016] Borowski, M. (2016). The infinite source of random sequences for classified cryptographic systems. *2016 International Conference on Military Communications and Information Systems, ICMCIS 2016*.

[Chandrakar, 2014] Chandrakar, S. (2014). An Innovative Approach for Implementation of One-Time Pads. volume 89, pages 35–37.

[Chaula, 2006] Chaula, J. A. (2006). *A Socio-Technical Analysis of Information Systems Security Assurance: A Case Study for Effective Assurance*. PhD thesis.

[Duncan and Whittington, 2014] Duncan, B. and Whittington, M. (2014). Compliance with Standards, Assurance and Audit: Does this Equal Security? In *Proc. 7th Int. Conf. Secur. Inf. Networks*, pages 77–84, Glasgow. ACM.

[Duncan and Whittington, 2016] Duncan, B. and Whittington, M. (2016). Enhancing Cloud Security and Privacy: The Power and the Weakness of the Audit Trail. In *Cloud Comput. 2016 Seventh Int. Conf. Cloud Comput. GRIDs, Virtualization*, number April, pages 125–130, Rome. IEEE.

[Duncan and Whittington, 2017] Duncan, B. and Whittington, M. (2017). Creating an Immutable Database for Secure Cloud Audit Trail and System Logging. In *Cloud Comput. 2017 Eighth Int. Conf. Cloud Comput. GRIDs, Virtualization*, pages 54–59, Athens. IARIA, ISBN: 978-1-61208-529-6.

[EU, 2012] EU (2012). Opinion 05/2012 on Cloud Computing (Data Protection).

[Guttman and Roback, 2011] Guttman, B. and Roback, E. A. (2011). NIST Special Publication 800-12. An Introduction to Computer Security: The NIST Handbook. Technical Report 800, NIST.

[Kerckhoffs, 1883] Kerckhoffs, A. (1883). La cryptographie militaire. *Journal des sciences militaires*, IX:5–83.

[Ko et al., 2011] Ko, R. K. L., Jagadpramana, P., Mowbray, M., Pearson, S., Kirchberg, M., Lee, B. S., and Liang, Q. (2011). TrustCloud: A Framework for Accountability and Trust in Cloud Computing. *Perspective*, pages 1–9.

[OED, 2016] OED (2016). Oxford English Dictionary.

[Pearson and Benameur, 2010] Pearson, S. and Benameur, A. (2010). Privacy, Security and Trust Issues Arising from Cloud Computing. *2010 IEEE Second Int. Conf. Cloud Comput. Technol. Sci.*, pages 693–702.

[Rijmenants, 2009] Rijmenants, D. (2009). Is One-time Pad History ? *Cipher Machines & Cryptology*, pages 1–4.

[Shannon, 1949] Shannon, C. E. (1949). Communication Theory of Secrecy Systems. *Bell System Technical Journal*, 28(4):656–715.

[Soares et al., 2014] Soares, L. F. B., Fernandes, D. a. B., Gomes, J. V., Freire, M. M., and Inácio, P. R. M. (2014). Security, Privacy and Trust in Cloud Systems. In Ko, R., editor, *Secur. Priv. Trust Cloud Syst.*, chapter Data Accou, pages 3–44. Springer.

[Tobin et al., 2017a] Tobin, P., Tobin, L., Blanquer, R. G., Mckeever, M., and Blackledge, J. (2017a). One-to-Cloud One-Time Pad Data encryption : Introducing Virtual Prototyping with PSpice.

[Tobin et al., 2016] Tobin, P., Tobin, L., Mc Keever, M., and Blackledge, J. (2016). Chaos-based cryptography for cloud computing. In *2016 27th Irish Signals and Systems Conference, ISSC 2016*.

[Tobin et al., 2017b] Tobin, P., Tobin, L., McKeever, M., and Blackledge, J. (2017b). On the Development of a One-Time Pad Generator for Personalising Cloud Security. In *Cloud Comput. 2017 Eighth Int. Conf. Cloud Comput. GRIDs, Virtualization*, pages 1–6, Athens, Greece.