digi.me

EXECUTIVE BRIEFING

Securing Personal Data

Dr Gavin Ray, CTO digi.me

EXPLAINING MODERN TECHNIQUES IN COMMERCIAL DATA PRIVACY PLATFORMS

Personal banking and health data have justifiably been secured in closed, vaulted systems for years, yet the online world has been dominated by multi-$Bn services from major technology brands, especially those free at the point of use, where personal data is often viewed as a tradable commodity.

Before we begin on a detailed insight into why digi.me is an authority on the integrity of security and privacy of personal data, it's key to highlight that digi.me's technology is built so that we [digi.me] never **see, hold nor touch user data** and that data integrity and privacy is at the very heart of everything we do. In this briefing we explain how both datasets are converging and evolving to adapt to the complex requirements of a user-centric and mobile computing world. It's been proven, time and time again, that liberating data and using it in new ways creates opportunities to offer new benefits to individuals, whether it's in specific areas like online banking, or generalised sectors like utility price comparison websites or even personalised shopping. There is an obvious need to ensure high security for all personal data yet support a flexible and fast-growing data-sharing economy, which must allow individuals to do more, not less, with their data. Data sharing is all about empowering and enriching individual's lives, as well as their experiences with the brands and service providers who employ the technology. The value comes from people seeing rich services based on multiple datasets from their personal life, but they need to be convinced that the process is secure and companies using it must conform to the new and emergent regulatory environment.

We explain the methods used in the digi.me privacy management platform to ensure personal banking, health and the whole range of online services' data is secured in the same way as in traditional closed systems. We then show how it can be made available in GDPR-class sharing services to the same standards, by the application of the latest cryptographic and security engineering methods to modernise the closed vault. This merger of old and new security architectures unlocks large-scale economic value, provides a far better breadth and quality of data and removes the burden of personal data storage from application and service providers.

Security for non-technical professionals

This briefing is written for a professional audience concerned with understanding the underlying mechanisms of security required to protect national health and finance data management technology. When setting out to secure personal data management systems, theory and practice are very different. Software must not only meet well defined security standards to determine things like the algorithms used to implement data encryption, it must address an entire spectrum of system security risks. Beyond the original 'intent' of a technology system design, one must always address the 'reality' of what happens between the inception and the execution of security standards. Secure system validation for national infrastructure is all about proving the entire system is protected in multiple ways against all eventualities.

The mantra of the digi.me solution is "we never see, hold, or touch your data", for this to be valid and valuable we must not only enforce protection of the systems so they operate as expected, but ensure that when processing user data they remain secure. We do this by dedicating processing functions to user data in a closed and tightly defined environment and wrapping protection around it.

The digi.me system has been shown to not only meet the required technical standards but has also been proven to have been implemented securely as a suite of mobile and desktop applications integrated with a cloud-based (online) service. Since this type of system is subject to a very wide variety of attacks from so-called 'bad actors' it was necessary for the digi.me engineering team to demonstrate to Health Authorities a wide and deep approach to securing all the various aspects of the software, its storage and its operation from the applications to the online systems. To define the security scope for compliance review in personal data environments we state seven core principles that must each be adhered to, throughout design, and to be proven in practice via intrusive analysis and aggressive physical validation. A secure system must be shown to meet and exceed the following primary requirements:

1. Storage Protection:
All user data-files are encrypted to, at least, banking standards and the AES-256 algorithm. This must use individual file encryption in addition to any encryption provided by the storage technology itself (i.e. cloud storage). To prevent a single decryption key from unlocking all data, there must be separate keys for every file.

2. Authentication & Authorisation:
All software components handling user-data (apps, systems and/or integrated services) must be able to prove their provenance, usually with a cryptographic certificate issued by a trusted party and impossible to forge. These credentials must be fully protected when not active and they must indicate the extent of the authorisation of the software to act under agreed commercial, statutory or regulatory instruments.

3. Transit Protection:
All user data must be secured in transit by encrypted connections with authentication of the sender and receiver, with no option for interlocution by bad actors able to forge credentials or exploit weaknesses in public internet standards, or published software.

4. Anti-Invasion:
The system must be protected against brute force, covert invasion and by the exploitation of any known vulnerabilities in public software components used in the construction of any of the applications or cloud processing systems.

5. Anti-Forgery:
All system data and credentials must be verifiable to prevent their forgery.

6. Chain-of-Trust:
All data entering the system must be provable as unaltered from first point of entry, moving between intermediate storage, transit services and any eventual recipients.

7. Integrity:
All code, systems and functions must be provable as original and constructed from verifiable sources to ensure system infiltration and corruption by bad actors cannot occur.

Examining the Security Objective

Security is all about proof

When we look at the security defined behind banking and health data storage there are a number of critical factors, each requires proof that they meet government defined standards. Data must be encrypted in ways that we know cannot be broken without inordinate effort, it must be stored in systems that do not allow unauthorized access and everyone accessing the data must have a means of proving they are who they say they are. It is important to understand the difference between theory and practice in this area.

By example, the digi.me framework uses industry leading encryption standards known as AES-256, which stands for the Advanced Encryption Standard from the US National Institute of Standards & Technology, and it uses 256-bit keys, which determines the mathematical complexity (hence time, cost and effort) of breaking the encrypted files.

Even low complexity software systems can say they use this standard, the code to run it is widely available. Yet, to be secure as a software implementation, the practical issue is that the engineers who use this standard can take this code from multiple sources – it is freely available as open source. The trouble with this is that software code is vast and itself packaged and shared from different places in different versions. To be certain of code quality and that there is no hidden “bad” code or a weakness in the coding itself it is necessary to only take code from known sources, to be certain that it has not been tampered with by using a “proof”. Good security code is available with a “signature” that allows us to prove it is original and not tampered with, so when building large systems all signatures must be checked. It’s also a truism in software that code that is new or recent, in the security space has risk. It takes a number of years for code relating to core security functions to be beaten to death in the industry, to survive multiple attacks from the public internet – which is in effect the battle field. Only when code is battle hardened is it good enough to use in encryption functions in the digi.me system or other banking and health solutions.

Industry technology standards state that checking a box to indicate a certain encryption method is used, is not enough to claim compliance. There must be evidence that the chain of software from core source components to the final product is proven and that when the system runs on the public internet it performs exactly as required. Infiltration of source code, corruption of the intervening systems and vulnerability to interdiction are all significant concerns. Defining and creating mitigation of all these practical integrity issues were the basis of the process we followed when gaining national infrastructure approval.

Security by design – A system focus

Using the simple AES-256 example we have exemplified the first principle of secu
rity, that it is a systems issue and security comes from a process of holistic design and detailed analysis of every risk from code errors to practical systems delivery and ultimately to proof by destructive testing. It is intuitive that security is an antidote to attack, what this means to a national infrastructure class privacy management solution like that of digi.me, is that one must define all the classes of attack and all the individual risks in order to defend against them.

To emphasize a point common to government, banking and health data security systems – security design must come from expertise and deep knowledge of the threat, so the only approach that works is to use experienced security professionals with advanced knowledge of the problem and build the software solutions with a team that understands national scale infrastructure.

The digi.me privacy management platform is based on design models derived from the security architects experience advising and validating finance, large enterprise and government systems and from the CTO experience in building global scale infrastructure. The system delivery is continuously monitored by deliberate attacks designed to emulate those in the real world and to ensure the established security levels are maintained. In proving a system is secure there are set of basic principles and critical elements that must be shown in all active systems. The list below shows how the entire flow of data and all the components within the flow must be secured and validated. It's worth seeing them summarised to understand the full scope of a system validation and how digi.me reaches the required standards:

1. Storage:
Data when stored must be encrypted with a proven, strong algorithm where - typically - the cost of attack is higher than the value of the data. AES-256 is the current, leading banking and health sector standard.

2. Storage:
When data is stored in a third-party provider's cloud service (like Microsoft OneDrive, GoogleDrive or Dropbox) even though it is encrypted by the provider, it must be double encrypted to ensure that compromise of the provider does not compromise the user data.

3. Storage:
All data files are individually encrypted to ensure that cracking encryption once, for one file, doesn't open up all data files to immediate access.

4. Authentication:
All user access to their own data must be authenticated, the user must prove they have the master password which acts as the key to their data and that the effort of guessing passwords is prohibitively high.

5. Authentication:
wherever user credentials to access their remote data are stored they must be encrypted to the maximum standard to avoid reverse engineering.

6. Authentication:
wherever the system connects to third party applications it must authenticate that the third party is the intended entity.

7. Authorisation:
wherever a user or third party application connects to the system it must only make authorised requests that can be irrefutably be validated and not forged.

8. Transit:
All data when transmitted must be encrypted and the encryption keys protected and validated.

9. Transit authentication:
When all data is transmitted it must be proven that each party has identified themselves uniquely.

10. Transit validation:
When all data is transmitted it must not be possible for a “man in the middle” to masquerade as an authorised recipient or transit entity for the user’s data.

11. Invasion:
At every point in a system there must be no means for an unauthorised third party to access user data or forge credentials to gain access, known as interdiction. This means protecting operating systems, their memory, their processing, their transmission functions and their storage.

12. Forgery:
Where user data is stored or shared it must not be possible for an interlocutor to modify the data anywhere from the mobile/desktop applications, via the networks or within any of the cloud processing, sharing or storage services.

13. Chain-of-trust:
Where data has entered the “system” it must be provable that the data has not been modified during storage or transit from first point of entry.

14. Source Integrity:
Where software is written to enact system functions it must be provable that the code is from “trusted” sources and not compromised.

15. Version Integrity:
where software is written to enact system functions it must be provable that the version deployed is the version written as intended and not modified by bad actors.

16. Proof:
Where the system enacts commands and moves data there must be proof of activity for the purpose of audit.

17. Privacy:
Where the system enacts commands and moves data there must be certainty that no user data or privacy artefacts are stored within the system. Battle hardened is it good enough to use in encryption functions in the digi.me system or other banking and health solutions.

Vaults only work if strong and the key is protected

There are many simple analogies that help us understand and explain software security, an obvious one is that a vault needs a key. So, no matter how strong the vault, if the key lies around, then the security is weak. Since it's obvious there are powerful demands for encryption in data storage and transit, there are equally strong demands to manage keys carefully. In a large-scale personal data and privacy management system it is mandatory that keys are carefully protected using the maximum extent of physical security common to modern smartphones. One of the most common concerns about online banking and similar services is that any keys required to access them are on a device that hackers may attack. Worse, a device is outside a traditional secure building loved by corporate security gurus, which means it can quietly and methodically be attacked away from prying eyes. Over the last few years major device manufacturers and operating system (OS) providers have made great strides in this area. To the extent that you can trust mobiles to hold keys securely once generated.

There is a further challenge with vault keys, relating to the simple concept of "how many attempts does it take to guess a 4-digit password". The obvious answer is 9999 + 1 for the 0000 option. So, security designers worry about so called brute-force attacks on secret keys, which is where hackers try every combination.

When looking at system security a major topic is key security and how it is protected, but without going into the mathematics of what remains a somewhat painful topic, it's important to use methods to take small simple, human friendly passwords and generate long random passwords. There are well-proven techniques that generate long randomised passwords from human inputs and this renders brute force attacks either moot or excessively expensive. The primary reason is that the computing cost of generating keys is very high and the maximum number of attempts to try them is rate-limited. A secure system must have significant focus on using proven, strong software in this area from high-confidence sources. Every element of the system touching and sharing keys is validated especially carefully in final production and constant monitoring and analysis of key usage must be used to ensure system integrity. This is one of the often hidden areas of system validation for good reason.

In theory the internet is secure and there are a great many good standards to allow it to be secure, but in practice the standards do not provide security if implemented weakly or with hidden mistakes. The root of the problem is that the mistakes that can be made are extremely easy for software teams to make if driven to write features and show demonstrations too quickly, or if the team's lack security specific training since the hidden issues in internet security are not the domain of the average engineer.

For anyone using a browser it is common to see the padlock symbol saying that the connection is secure. This comes from a universal standard called HTTPS, which is derived from its name HyperText Transfer Protocol – Secure, as derived from Tim Berners-Lee original 'father of the internet' invention. This technology is based on a number of relatively arcane functions to do with how bits and bytes are moved over wires, but it also uses a certificate of authenticity to allow a website to say "I really am who I say I am".

There really is a minefield of errors and practical challenges in implementing this process, where even small mistakes can lead you to think you are in a secure communication link, but someone is sitting as the "man in the middle", one of the biggest threats to both public internet security and also wi-fi networks. Think of it like this, if someone were to stand in front of your house and pretend to be busy preening the pathway, the postman walks up and imagines they are the homeowner and hands over the post with birthday cards and cash gifts inside. This is the same principle of masquerading on the internet. The logical answer is for the postman to ask for an identity card before handing over the post and for the house-owner to ask for ID that the postman is real and that the post

isn't something dangerous.

In proving that an online personal data and privacy service is good enough to handle medical and banking data, it is essential to demonstrate with real proof that every version of this Man-in-the-Middle attack cannot happen at every point in the solution. This is especially key where data is shared from service providers to users, users to data-consumers and all the business processes that may go via the internet.

As before, proving this requires that all the code is from reliable safe sources, that all the functions work and in the case of the verification that digi.me has undergone, that multiple attacks from every direction are rejected.

Implementing Privacy Mandates

To end on the topic of privacy and the confluence of traditional high-security health and banking systems with modern online services, it is essential to show how legal and regulatory models are enforced within the technology.

Much as a burglar alarm lets you know there is an intruder but does not fight back; system security and privacy controls are only about protection but there will always be a point where the real world meets the logical world of software. In the event of disputes, or fear of dispute, the technology solution must integrate with regulatory and legal frameworks by generating auditable proof of relevant actions.

What this means in practice, for privacy management systems, is that there must be electronic contracts and log events that show when a contract was created, when it was agreed, when it was enacted. The contracts must be protected so they cannot be forged or changed, the logging records must relate to the user as an entity, but they must not store personally identifiable data, save for the need to show a user exists and what actions were taken on their behalf. There must be sufficient proof and verification in the logging functions to allow a clear binding of legal/regulatory activity to matters of fact.

Conclusion

When assessing the suitability of a privacy management platform for use in the storage and control of personal data including finance, health and other sensitive datasets it is necessary to view the depth of protection from both a theoretic and practical sense. Assessment of the entire design must be allied to practical verification of all the major threats. Despite the use of encryption and internet security, the important factors include in-service software validation and proof that the original code is protected. In running a privacy management service for the general user, the latest statutory and regulatory controls are powerful and effective yet deserve proof of adherence. A system must be secure as a service, but it must also provide operational proof of its internal function and that of the service it delivers. Digi.me has spent three years investing on the most advanced privacy engineering systems and has been defining, then proving best practice.

CONTACT

Address

The Old Coach House, Grange Court,
Grange Road, Tongham, Farnham,
Surrey, GU10 1DW UK

Phone & Fax

Phone: +44 (0) 1252-781533

Online

Email: info@digi.me
Website: https://digi.me