absio®

Implementing the Least-Cost, Least-Time Solution for GDPR-Compliant Software

720 836 1222 **sales / support**

sales@absio.com **email**
www.absio.com **web**

8740 Lucent Boulevard, Ste 101
Highlands Ranch, CO, 80129

SUMMARY

Using Absio's toolset, with a few simple lines of code, application developers can create GDPR-compliant software that:

- Automatically, individually encrypts any type of unstructured data generated or processed by an application everywhere it exists without having to manage keys, add hardware, increase latency, or rely on a third-party service for access to data
- Extends automatic PKI cryptographic authentication and encrypted data exchange to any user—inside or outside your network
- Automatically assures TLS for every transmission
- Tracks, updates or deletes a data subject's content everywhere it exists per a schedule or on-demand
- Cryptographically binds metadata from any source to unstructured data dictating how the data can be accessed and used
- Integrates with other information systems - providing decrypted content anywhere it's needed for analysis and processing
- Enables provable GDPR compliance

Data controllers and processors—and the software providers and developers they must necessarily rely upon to become and remain compliant—are facing a unique challenge. Unlike other cybersecurity and privacy regulations, the GDPR does not define compliance methods. It specifies nothing about tools, processes, or technical standards, nor does it provide templates, guidelines, frameworks, examples, or best practices.

Instead, GDPR prescribes specific forms of multi-party (controllers, processors, and data subjects) control over personal data, which must be maintained from the moment the data is gathered until the moment it ceases to exist. The GDPR also dictates potentially severe penalties and damages for data processors and controllers who fail to adequately control the personal data with which they've been entrusted.

The personal data that processors and controllers must control is stored in two forms—structured and unstructured. Structured data is stored in and processed by certain kinds of databases. Unstructured data is stored as files, that is, discrete digital objects such as documents, images, videos, audio, forms data gathered and stored for later inclusion in a database, and data extracted from a database and stored as files for subsequent use.

Maintaining control of individual files in their native form is all but impossible. Files are purposely designed for unrestricted reading, copying, distribution, alteration, and deletion, anywhere, at any time, by anyone or anything that has an instance of software capable of opening and operating the file. The abuse of security and privacy that results from failing to control files is a central problem the GDPR seeks to correct.

Attempts to control files can be either indirect or direct. Indirect control takes multiple forms, but in general it seeks to control files sometime after the files are created using a combination of specialized analysis software and a combination of human and software-enforced policies. Examples include

running file discovery tools to find regulated content, moving the subset of files with regulated content to designated storage locations that are secured, adhering to internal and external distribution protocols, following administrative constraints on how files are used, and executing data retention policies—hopefully all without error. The indirect approach tends to be expensive because it is labor-intensive, and risky because its reliance on humans produces inconsistent results. Furthermore, the indirect approach will be increasingly challenged by the vast and accelerating amount of unstructured data being gathered and processed by each new and more powerful software application, and the rise of NoSQL big data processing against unstructured data. In simple terms, the multiple systems and protocols inherent in the indirect approach cannot "keep up" with the amount of data being created and the number of locations where it may be located (cloud, on-premise, mobile, IoT, etc.).

Direct control, on the other hand, ensures that all data generated or processed by software applications is automatically controlled prior to it being stored or distributed. Direct control requires upfront development work to build new or modify existing applications that generate or process regulated data, but eliminates the need to find and control data once stored in any number of known and unknown locations. Direct control provides a path towards complete data control, whereas indirect does not. New data may not even be indirectly controlled for some time.

HOW ABSIO TECHNOLOGY DIRECTLY CONTROLS DATA

Absio offers a new kind of Serverless Encryption™ technology that enables software applications to directly control data throughout its existence. It is a highly-automated data containerization toolset that provides pre-built interconnected functions that can be incorporated by software developers into existing or new applications with a few simple lines of code.

The toolset consists of multi-language software development kits (SDKs), each with a simple application programming interface (API), and an optional server application (Absio Broker™) that can be installed on premise or in the cloud. Absio technology functions across platforms and devices.

The Absio Data Encryption SDKs generate keys for PKI-based authentication and data-level encryption. All key generation, encrypt and decrypt processes happen on the device, server, or browser running the Absio-enabled application (an application that has incorporated the Absio SDK) without calling a central server. Key generation and management is automatic, so there is no key-related administrative burden and developers need no cryptography expertise. Encryption and decryption does not require a connection to the internet, or investment in key servers, hardware security modules, or appliances.

The Absio SDK generates private and public keys for all users (human or system) of an Absio-enabled application used for signing and derivation. When the SDK receives data from an Absio-enabled application, it creates an encrypted data container (called an Absio Secure Container or ASC) and assigns it a global unique identifier (GUID) that enables it to be tracked, updated, and deleted, in and out of network, throughout its existence. The SDK encrypts the data content, and, if applicable, adds any associated metadata (information about the content or controls regarding its use) to form the ASC. The metadata can either be encrypted and bound to the data, or stored separately for rapid identification and processing without providing access to the content itself. The SDK individually encrypts each ASC with its own unique key. ASCs can be decrypted when and where needed, but by default, are only decrypted in memory while the content is in use. Data in storage and in transit is always encrypted.

For data in use, the SDKs can enable persistent control of how, where, and when decrypted content can be used. In other words, even after the data is decrypted, it can only be accessed or used in ways specified by the developer. Metadata of any type, from any source, can be cryptographically bound to data content. When the ASC is accessed by an authenticated user, the Absio-enabled application consumes the ASC's metadata and only permits use of the decrypted content in accordance with instructions contained in the metadata.

The Absio Broker application, in conjunction with the SDKs, can provide cryptographic user authentication, management of public keys, encrypted file storage, encrypted file sharing, transmission encryption, and encrypted file backup. The Absio Broker application, by default, is "blind." Content and keys are encrypted by the SDK prior to transmission to the Absio Broker application. No trust relationship with a public or hybrid cloud provider is required, and accidental or malicious exfiltration of content and keys stored by the Absio Broker Application yields no usable information.

HOW ABSIO TECHNOLOGY MEETS GDPR REQUIREMENTS

Encryption, Integrity, and Confidentiality[1]

Absio-enabled applications automatically encrypt new data as it is created and existing unencrypted data as it is processed. Once encrypted, data is protected in storage and in transit, and is only decrypted in memory while in use by a cryptographically-authenticated user. Each application user is automatically assigned a unique user ID (UUID). Private signing and derivation keys for each user are automatically created by the SDK, encrypted, and stored in a key file on the device or in the browser running the application. Compromised user credentials cannot be used to decrypt content on devices or in browsers if the key file is not present.

Absio-enabled applications communicate with the Absio Broker application via a TLS-encrypted connection. Since data is encrypted prior to transmission, a compromised connection or breach of files stored by the Absio Broker application yields no usable content. Replay attacks are prevented by one-time-use, signed, time-sensitive session tokens. Attempts to reuse tokens or to use tampered-with tokens are rejected by the Absio Broker application. Users authenticate with the Absio Broker application by signing with their private key, mitigating the possibility of man-in-the-middle attacks.

Data content and keys are tamper-resistant. Data integrity is verified via hash-based message authentication code (HMAC) prior to decryption, and all keys are signed with the user's private key to provide source verification.

Pseudonymization and Anonymization[2]

For purposes of pseudonymization, developers can use the Absio SDK to hash data and make the hash map available only to select cryptographically-authenticated users. The hash map is stored in an individually-encrypted ASC, so that only users with the appropriate permissions can access the ASC and decrypt the map. The SDK does not provide means to anonymize data, however, anonymized or pseudonymized data can also be encrypted if desired.

Electronic Access

The GDPR grants data subjects a wide range of access and personal data control rights, each with a common requirement: electronic access. Absio-enabled applications can extend cryptographic authentication to any user, including data subjects. Additionally, other methods of user authentication are supported. For example, Absio-enabled applications can consume metadata from other systems, such as file discovery and analysis systems, data governance systems, identity access and management (IDAM) providers (e.g., Active Directory and LDAP), or multifactor authentication (MFA) providers.

Right to Update and Delete Personal Data[3]

Due to the inherent uncontrollability of unstructured data, when a data subject, controller, or processor wishes to update, suspend the processing of, or delete files containing personal information, the challenge can be knowing with certainty where all the copies of a given file are located, and which files contain the information that needs to be updated. Each ASC generated by an Absio-enabled application has a GUID that can be identified and tracked when accessed by an Absio-enabled application. Additionally, classification, labeling, and tagging metadata can be bound to data and processed by Absio-enabled applications, making identifying and updating content and metadata information simpler.

Data Lifespan Control[4]

Using Absio-enabled applications, data subjects and controllers can define a data object's lifespan, determining how long data content is available. Lifespan metadata can be cryptographically bound to data ensuring that obsolete data is no longer accessible and permanently deleted everywhere at a predetermined time. Access to shared data can also be revoked or modified on demand.

Sharing Data with Other Controllers[5]

Absio-enabled applications allow data to be shared between controllers and/or processors without losing control of the content. An application provided by the sharing controller/processor and deployed by the receiving controller/processor, receives the encrypted content and then either decrypts that content upon its arrival or stores it in encrypted form and then decrypts it on-demand. This enables the sharing controller/processor to document that shared content was always encrypted while it was in their possession and when it was shared, and with whom and when it was shared. The sharing controller/processor can also apply policy metadata to any information shared dictating how the data content can be used and with whom it can be shared when accessed by the Absio-enabled application.

Privacy By Design and By Default[6]

Use of Absio technology is a clear indicator of compliance with the “technical measures” of GDPR privacy by design and by default requirements. Absio technology is purposely designed by default to protect all the data it handles and to extend control of access, use, and lifespan of data to the data subjects themselves in addition to controllers and processors.

Demonstration of Accountability[7]

The use of Absio technology itself can demonstrate compliance with data protection and other applicable requirements. If demonstrable continuous compliance is desired, applications can create a log of all relevant Absio-enabled functions performed by the application, such as ASC creation, access, sharing, updating and deletion; user creation and deletion, logins and logouts; and time, sources, and destinations of shared data for auditing purposes.

Data Residency

The Absio Broker application is portable and can be installed where needed to conform to data residency requirements.

HOW ABSIO TECHNOLOGY CAN REDUCE THE COST OF GDPR COMPLIANCE

Buy or Build

By offering a complete, low-code, cross-platform toolset for securely storing, transmitting and sharing data, Absio can save man-years of software development time and reduce investments in additional software, hardware and IT resources.

Modularity and Incremental Implementation

Absio technology is intentionally modular. Data encryption, cryptographic authentication, encrypted data sharing, and metadata-based data control can all be accomplished separately. Depending on how a given application is architected, only certain Absio capabilities may be needed. In other cases, Absio capabilities can be incrementally implemented over time based on business priorities, budget, and the availability of developer resources.

Interoperability

Absio technology is designed to integrate with and extend the capabilities of other systems, not replace them. Absio-enabled applications can securely deliver data in its native format to systems designed to consume them. Examples include threat mitigation systems, data loss prevention systems, archiving systems, or any other system that processes or analyzes data. Absio-enabled applications can consume information provided by data governance systems, file discovery and analysis systems, policy engines, or any other system that produces metadata intended to manage data, then return the information to the application when needed.

Risk and Liability Mitigation[8]

Absio provides provable, continuous encryption of data in motion and at rest, including data shared with other controllers and processors. It is unlikely that Absio-protected data will incur penalties or damages due to a data breach, and the use of Absio technology may reduce cyber insurance costs.

Reduced Data Protection and Management Cost

The use of Absio technology may reduce or eliminate certain data protection and data management costs. For example, an organization may have information assurance personnel, specialized applications, and administrative processes dedicated to finding files containing personal information, moving them to a specific location, and then encrypting them. It may be less expensive to simply use Absio tools to automatically encrypt all newly created and existing files in storage, and leave them where they are.

CONCLUSION

Absio technology may well be the least-cost, least-time GDPR compliance solution for your organization. For more information on the Absio toolset, pricing, or to arrange a free trial, please contact us at sales@absio.com.

---

1. Data is secured, and integrity and confidentiality are maintained, using technical and organizational means under the management of the controller: Recital 49 and Articles 5-1(f), 32-1(b-d). Data encryption shall be used, when possible: Recitals 83 and Articles 6-4(e), 32-1(a).

2. Data pseudonymization shall be used, when possible: Recitals 26, 28, 29, 78 and Articles 6-4(e), 25-1, 32-1(a). Data shall be anonymized, when possible: Recital 26.

3. The data subject shall have the right to have their data updated, free of charge, if there is an error: Recitals 59, 65 and Article 16, and, the data subject shall have the right to request this update electronically, Recital 59.The data subject shall have the right to have their data erased without undue delay: Recitals 59, 65 and Articles 13-2(b), 14-2(b), 17, and, the data subject shall have the right to request this deletion electronically, Recital 59.

4. Data is stored only for the time necessary to meet the objectives of the data subject. Out-of-date personal data shall not be stored. (Part of an Electronic Records Management strategy). And the data subject shall be notified of this time period or its calculation approach at the time of the data capture: Recitals 39, 45 and Articles 13-2(a), 14-2(a), 25-2

5. The data controller must notify other IT organizations that hold the data subject's data that the data subject has requested data erasure: Recital 66 and Article 19 (Therefore, the IT department must know where all the data subjects' data is being stored by third parties so that these third parties can be notified of erasure requests. Up-to-date internal and external data inventories are critical.)

6. Implementing data protection in the system and the organization, by design and by default, is a legal requirement: Recital 78 and Article 25

7. Recital: 39-2. The controller shall be responsible for, and be able to demonstrate compliance with, paragraph 1 ('accountability').

8. Article 82-3. A controller or processor shall be exempt from liability under paragraph 2 if it proves that it is not in any way responsible for the event giving rise to the damage. Article 34-3 The communication to the data subject referred to in paragraph 1 shall not be required if any of the following conditions are met: (a) the controller has implemented appropriate technical and organisational protection measures, and those measures were applied to the personal data affected by the personal data breach, in particular those that render the personal data unintelligible to any person who is not authorised to access it, such as encryption.