CS INSTRUMENTS

CS Leak Reporter Cloud Solution
Cloud data storage security concept

1 Table of content

1 Table of content 2
2 Preface 3
3 Cloud Solution 4
4 Data security 4
4.1 CS INSTRUMENTS Guarantee 4
4.2 Proven cloud infrastructure 4
4.3 State of the Art Identity and Access Management (IAM) 4
4.4 Access to data only via protected service APIs 5
4.5 Isolation of data in organizations/realms 5
4.6 Fine granular access permissions within the organization 5
4.7 Modern software architecture 5
4.8 Minimum number of service accounts 5
4.9 Regular review of the logs 5
4.10 Access only via https (TLS) 6
4.11 Data economy 6
4.12 On-Premise solution (installation by the customer) 7
4.13 Comparison 7

2 Preface

The Leak-Reporter Cloud application consists of a set of software components that realize multiple users' management of leakage data. The parts are divided into backend components/services that run on the server and communicate with the database and the front-end applications provided by a web server and run in the browser. Thus, the user can easily use the application via the browser, and no application needs to be installed on the system (PC, tablet).

The installation of the components can be done on the customer's server (on-premise). On the other hand, CS INSTRUMENTS offers the much more modern approach of a cloud solution, where the application runs on servers of CS INSTRUMENTS. See Fig. 1.

***Figure 1**: Multi-user leak management with CS Instruments Cloud or On-Premise installation of Leak Reporter software components.*

3 Cloud Solution

CS INSTRUMENTS has installed the components on a cloud platform and offers them to customers for use. This has the enormous advantage that the customer has no effort to install and maintain the system. In addition, the responsibility for maintaining the software infrastructure lies with CS Instruments. Updates for troubleshooting or providing extended functions are installed centrally on the cloud servers and are available to the end-users immediately. In addition, more applications and features will be added to the cloud solution in the future. By storing all data on multiple cloud servers, local hardware and software issues are also less likely to result in data loss and provide higher availability.

4 Data security

All customers use the software, so the data is also stored together. Of course, one customer is not allowed to see another customer's data. This is ensured via the integrated identity and access management system (IAM). In addition, however, the APIs are publicly available (and therefore vulnerable), and the server is not under the control of the individual customer. Accordingly, data privacy and data security are of extreme importance.

The following are, therefore, essential points about data protection:

4.1 CS INSTRUMENTS Guarantee

CS INSTRUMENTS contractually guarantees that no data will be forwarded to third parties and will only be used in case of errors to fix bugs and thus improve the product for all users.

4.2 Proven cloud infrastructure

The software is based on the infrastructure of an experienced cloud provider (Microsoft Azure) with appropriate security measures. According to the applicable European Data Protection Directives, the data is stored in Western Europe.

4.3 State of the Art Identity and Access Management (IAM)

- Authentication and authorization are performed via a widely used open-source IAM system, which is regularly updated.

- Only the key derived from the user's password corresponding to PBKDF2 is stored in the user database (there is no way to reconstruct the password).

- Login can be done only through the login page provided by IAM (which makes phishing attacks with imitated login pages more difficult). For authentication, OpenID-connect with the authorization code Flow is used (https://openid.net/connect/). The user/client (browser) then contains access and refresh token (see https://oauth.net/2/). The tokens that are stored in the browser (as cookies) (for Single Sign-On) do not contain any access data and are only valid for a limited time (inactivity blocking).

4.4 Access to data only via protected service APIs

The databases, secured with username and password, are not accessible from the outside (public networks). Instead, the data is made available via APIs by services that run together with the databases in Kubernetes. In order to check the user's permissions, the client must send along the access token received from the IAM. The backend service checks this for validity (signature of the token and public key of the IAM) and then the authorization for the desired data is requested from the IAM.

4.5 Isolation of data in organizations/realms

The IAM ensures that users only see data (other users and leakage data) from the organization (realms) to which they belong (data from other organizations cannot be viewed).

4.6 Fine granular access permissions within the organization

Users themselves (starting from the organization administrator) can configure and assign individual access rights to data within the organization. In this way, external users can also be allowed to view (and, if necessary, edit) certain data. The administrator can revoke granted permissions at any time with virtually immediate effect.

4.7 Modern software architecture

The application was implemented in the form of a microservice architecture. This allows to reduce the complexity of individual components and thus increase security. In addition, technologies were used that make, e.g. SQL injections, impossible.

4.8 Minimum number of service accounts

Access to Kubernetes as well as the Azure APIs / portal is only possible for a few service accounts (persons). This also applies to the database backups and log files stored in Azure.

4.9 Regular review of the logs

Irregularities that indicate potential attacks or unauthorized access are detected early, and necessary countermeasures can be taken as quickly as possible.

4.10 Access only via https (TLS)

Access with http is not possible. All communication between client and server is encrypted, data integrity is ensured, and man-in-the-middle attacks are not possible.

4.11 Data economy

- Only the data required for the application is stored. This includes the user data as well as the leakage data.
- The database and log files are only stored for a certain period and then completely deleted.
- No tracking data of the users is collected.

The rough software access and security concept that was implemented for the CS Cloud solution is outlined in **Figure 2:**

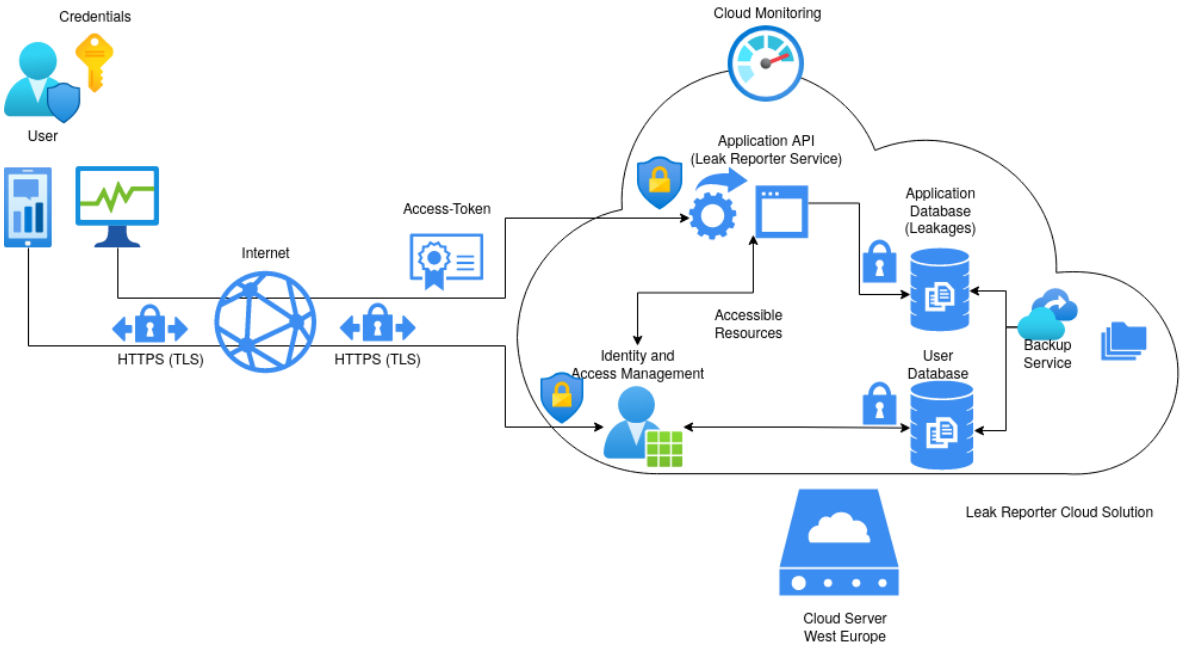

Figure 2: *Concept of the CS Cloud solution regarding security. Above all, the integrated identity and rights/access management ensures that each user only sees the data intended for him.*

4.12 On-Premise solution (installation by the customer)

With the On-Premise solution, the software components provided by CS INSTRUMENTS are installed by the customer. Although CS INSTRUMENTS offers a compact procedure for this in the form of a Docker Compose file, the installation and configuration nevertheless requires a great deal of time and demands extensive IT expertise from the customer. For example, the customer or its IT must take care of encrypted communication (https) or backup mechanisms. Furthermore, the customer has to take care of maintenance and updates. This is even more true if the installation takes place in a Kubernetes cluster (directly at the customer's site or at a cloud provider).

CS INSTRUMENTS has large personnel resources for development, operation and maintenance and can therefore complete the tasks in the cloud solution faster than employees of the company's own IT department, who often must pursue and prioritize other urgent projects in parallel.

4.13 Comparison

The following table compares some features of the approaches:

Table 1: *Comparison of different features when using the CS Cloud solution and an On Premise installation.*

Features	**CS Cloud Solution**	**On Premise (installed by the customer)**	
		Individual cloud servers	**Servers in the company network**
ownership of the servers used	✗	✗	✓
Security	High	High	Extremely high
Availability/redundancy	High (Kubernetes)	Low – high possible	Low
Accessible everywhere	✓	✓	✗
Cooperation with external service providers or customers	✓	✓	✗
Installation Effort	-	High	High
Initial costs	Low	High	High
Maintenance Effort	-	High	Medium
Running costs	Low	High	Medium