White paper

Secure file sharing and collaboration in the OpenText Cloud

An overview of security features in OpenText Core

Security is paramount when it comes to sharing an organization's most valuable asset in the cloud—data. This white paper discusses the security features built into OpenText™ Core, as well as in the OpenText Cloud generally.

Contents

Secure, dedicated data centers for North America and EMEA 3

Secure file encryption in transit 3

Secure file encryption at rest 3

Multi-tenancy: Ensuring secure separation of tenants and data 4

User permissions control 4

Audit trails and version history 4

Tenant administration and user control 4

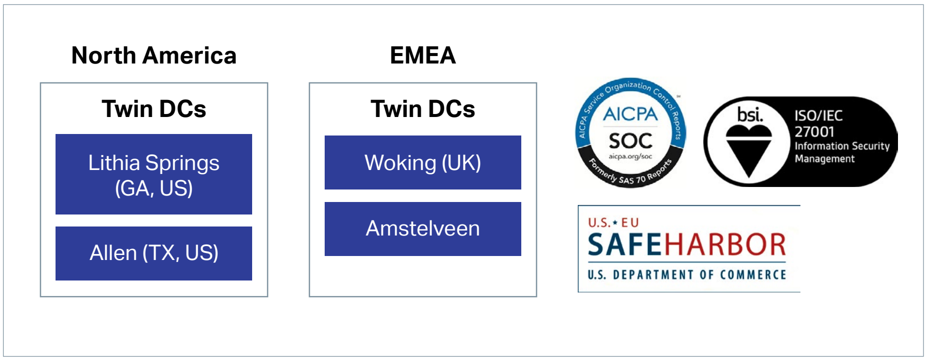
Public links and external sharing 4

Two-factor authentication 4

Single sign-on (SSO) 5

Secure, dedicated data centers for North America and EMEA

Enterprises trust OpenText with their business-critical information, in large part due to its commitment and expertise in cloud security, privacy and trust. OpenText owns and operates its own data centers and infrastructure around the world. OpenText Core runs on twin data center regional pairs to address data sovereignty requirements and performance expectations globally.



North America

Twin DCs

Lithia Springs (GA, US)

Allen (TX, US)

EMEA

Twin DCs

Woking (UK)

Amstelveen

AICPA
SOC
aicpa.org/soc
AICPA Service Organization Control Reports
Formerly SAS 70 Reports
bsi.
ISO/IEC 27001
Information Security Management
U.S. • EU
SAFEHARBOR
U.S. DEPARTMENT OF COMMERCE

Each twin data center pair maintains SOC 1 Type II, SOC 2 Type II and ISO27001: 2013 certification.

Secure file encryption in transit

Transport layer security (TLS) is leveraged for file encryption in transit. The benefits of TLS include strong authentication, message privacy and integrity, enabling the detection of message tampering, interception and forgery.

Secure file encryption at rest

Files are stored in distributed data centers with unique encryption keys for each client. The OpenText encryption solution leverages hardware security modules (HSMs) with tamper-proof backups, allowing for physically secure file storage and data loss prevention.

Upon the onboarding of a customer to OpenText Core, a 256-bit AES symmetric key encryption key (KEK) is generated within a secure hardware security module appliance. The HSM is a vault—the key can never leave the appliance, except to replicate to additional HSMs for scale and to back up HSM appliances. This ensures that the customer's KEK is kept completely confidential.

When an employee logs into OpenText Core and proceeds to upload a document, the file is streamed to an encryption service that generates a 256-bit AES symmetric data key (DK) and encrypts the document using the DK. At this point, several operations kick off:

1. The encrypted document is saved to the data centers local back-end storage facility and will be replicated to a peering site nearly immediately.
2. The data key used to encrypt the document is sent to the key management service (KMS), which interacts with the HSMs. The KMS sends the DK to the appliance, requesting that the HSM encrypt the DK using the customers KEK.
3. Upon receiving the encrypted data key (EDK), it is stored as metadata in the OpenText Core service and the DK is deleted from the server's memory.

Multi-tenancy: Ensuring secure separation of tenants and data

The multi-tenant architecture of OpenText Core allows OpenText to provide a best of breed enterprise information management (EIM) solution, leveraging a common platform for shared resources and economies of scale. Each tenant of the service is provisioned a unique space in the Identity Management service and back-end storage. The data model supporting the tenants builds relationships to manage the tenants' entitlements, users, files, comments, shares and metadata. The OpenText Core service provides functionality through restful APIs which are used by web, mobile and desktop clients. Every request performed against the API services is checked to ensure the user is authenticated and has the necessary authorization to access to the requested resource or function.

User permissions control

Read, write and delete action availability is set for individual users at individual file and folder levels with cascading permissions that allow for collaboration with internal and external parties.

Additional permissions information:
https://core.opentext.com/support/discussion/121/tip-sharing-permissions

Audit trails and version history

Audit trails keep activity records for files and their metadata. An audit trail provides assurance of the integrity of the electronic record. OpenText Core provides full audit trail capabilities, including file and folder audit history, version history and an activity feed with a timeline view of what has transpired to maintain the integrity of the file.

Tenant administration and user control

Organizations can administer a group of users, called a tenant. Administrators can lock, disable and revoke sessions and shares for users in their tenant. Tenant security settings provide administrators with control over password enforcement, complexity and session policy, as well as full visibility into sharing activity. Control is extended to mobile users and if a device is lost or stolen, the administrator can remotely wipe the device.

Public links and external sharing

For published files accessed via web link, expiry dates and password protection allow for safe sharing outside of the client's organization. Administrators have full policy control over mandatory complexity of passwords and date expiry.

Two-factor authentication

Two-factor authentication enables an additional authentication layer when accessing OpenText Core. Providers are extensible and include time-based, one-time passwords or integration with Duo premium services leveraging OpenText™ Directory Services. Granular settings, such as remembering devices and controlling account recovery, provide flexible levels of security to meet client requirements.

Additional two-factor authentication information:
https://core.opentext.com/blog/wp-content/uploads/2016/12/2FA-1.jpg
https://core.opentext.com/blog/wp-content/uploads/2016/12/2FA-2.jpg

opentext™

- See the demo
- Join the conversation
- Keep up to date
- Support

Single sign-on (SSO)

SSO allows an organization to leverage its existing enterprise directory infrastructure so that users are automatically created in OpenText Core and can sign on using their Microsoft® Active Directory® or LDAP credentials.

Changes to the directory are automatically pushed to OpenText Core, ensuring that access to sensitive files is reflective of organizational changes. Using SSO ensures that sessions are provisioned according to the organization's existing sign on policy.

Additional SSO information:
https://core.opentext.com/support/discussion/153/about-single-sign-on-and-opentext-core#latest
https://core.opentext.com/support/discussion/152/part-1-authentication-setup#latest

About OpenText

OpenText, The Information Company, enables organizations to gain insight through market leading information management solutions, on-premises or in the cloud. For more information about OpenText (NASDAQ: OTEX, TSX: OTEX) visit: opentext.com.

Connect with us:

- OpenText CEO Mark Barrenechea's blog
- Twitter | LinkedIn

opentext.com/contact