SAP Ariba

Removal of weak TLS 1.2 ciphers and adding support for TLS 1.3

Chitranjali Banjare
October, 2024

Agenda

1. Details about TLS changes and timeline

2. Impact on customers/Partners/Buyers/Suppliers

3. Actions Required

4. Verify and Test your TLS changes

5. FAQs

6. Q&A

TLS upgrade and Timeline

Starting January 24, 2025 **SAP Ariba and SAP Business Network** will:

- ➢ Start enabling support for TLS 1.3 connections.
- ➢ End support for weak TLS 1.2 ciphers while continuing support for strong TLS 1.2 ciphers

- ➢ Changes will be deployed Data center wise **and it will take until April 30, 2025 to complete the roll out across all our Data centers.**
- ➢ The change is applicable for both inbound and outbound connections
- ➢ Removal of weak TLS 1.2 ciphers and adding TLS 1.3 is essential for enhancing SAP Ariba and SAP Business Network security. Weak ciphers can be exploited by attackers to decrypt sensitive data, perform man-in-the-middle attacks, or compromise the integrity of communications. While, TLS 1.3 eliminates outdated and vulnerable cryptographic algorithms and protocols, making it more secure against known attacks compared to previous versions.

TLS upgrade – Technical details

Enabling Support for TLS 1.3

SAP Ariba and Business Network will continue to support TLS 1.2. In addition, we will include support for TLS 1.3 as well.

Following TLS 1.3 ciphers will be supported:

- TLS_AES_256_GCM_SHA384
- TLS_CHACHA20_POLY1305_SHA256
- TLS_AES_128_GCM_SHA256

Continue Supporting following TLS 1.2 ciphers that we recommend you to add:

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384

TLS upgrade – Technical details

Disable support for weak TLS 1.2 connections

SAP Ariba and Business Network will disable support for following types of ciphers:

1. CBC mode ciphers

Examples of CBC mode ciphers:

- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256

2. SHA-1 hashing based ciphers.

Examples of SHA-1 hash based ciphers:

- TLS_RSA_WITH_CAMELLIA_256_CBC_SHA
- AES128-SHA

3. RSA Key Exchanges based ciphers.

Examples of RSA key exchange based ciphers:

- TLS_RSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_NULL_SHA256

Impact on Customers/Partners/Buyers/Suppliers

Starting January 24, 2025, the following will happen :

For systems using only weak TLS 1.2 cipher suites:

- TLS handshakes will fail.
- Secured connections will not be established.
- All access to applications and network will not be available.
- All functionalities, released previously or in the future, will not work or be available.

For systems having both weak TLS 1.2 cipher (CBC, SHA-1, RSA Key exchange) and our recommended TLS 1.2 cipher suites enabled:

- TLS handshakes will automatically use the strong TLS 1.2 connections. We suggest you support our recommended TLS 1.2 ciphers and remove weak TLS 1.2 ciphers.

Actions Required

- Add support for following strong TLS 1.2 cipher suites as soon as possible for improved security:
  - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
  - TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
  - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
  - TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
  - TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
  - TLS_DHE_RSA_WITH_AES_256_GCM_SHA384

- After enabling strong TLS 1.2 cipher suites, or if you are already using strong TLS 1.2, you should remove all weak TLS 1.2 cipher suites.
- In addition, we strongly recommend enabling TLS 1.3 which is the current protocol that provides maximum security connections. We suggest using at least one of the following TLS 1.3 cipher suites:
  - TLS_AES_256_GCM_SHA384
  - TLS_CHACHA20_POLY1305_SHA256
  - TLS_AES_128_GCM_SHA256

Notes

- **Do not remove weak TLS 1.2 cipher suites unless one or more of the strong TLS 1.2 cipher suites have been enabled in your system.**

- **Support both TLS 1.3 and TLS 1.2 (with our recommended ciphers) till we release the TLS changes in all our Data centers i.e. by April 30, 2025. i.e. Do not remove TLS 1.2 till our changes are deployed completely.**

Verify and Test your TLS changes

- Customers/Partners/Suppliers/Buyers can VERIFY and TEST their TLS 1.2 connections using our custom domain test URLs.

- Test URL for TLS 1.2 protocol: Enabled with TLS1.2 ciphers only: https://tls12-strong-cipher-check.xglab.ariba.com

Supported Cipher list:

- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES128-GCM-SHA256
- ECDHE-ECDSA-AES256-GCM-SHA384
- DHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES256-GCM-SHA384

- Test URL for TLS 1.3 protocol: Enabled with TLS 1.3 ciphers only: https://tls13-strong-cipher-check.xglab.ariba.com/

NOTE: These test URLs are intended solely for evaluating TLS protocol and cipher compatibility and should not be used for end-to-end testing.

FAQs

➢ How to add support for TLS 1.3?

The process/steps for updating the TLS protocol and ciphers varies for different tools and libraries. **Following are few examples**:-

For Browser:

Upgrade to the latest browser or minimum following browser version which by default support TLS 1.3 and strong TLS 1.2 ciphers:

- Google Chrome (88 or above)
- Microsoft Edge (88 or above)
- Mozilla Firefox (87 or above)
- Apple Safari (15 or above)
- Mobile Safari on iPad (15 or above)

For JAVA client:

TLS 1.3 will be enabled by default on the client for JDK 8. The TLS 1.3 protocol can be enabled on the client using the jdk.tls.client.protocols system property, for example:

```
java -Djdk.tls.client.protocols="TLSv1.3,TLSv1.2" ...
```

or by using the https.protocols system property if the application is using the HttpsURLConnection or URL.openStream() APIs, for example:

```
java -Dhttps.protocols="TLSv1.3,TLSv1.2"
```

FAQs

➢ How to add/remove TLS cipher suites?

The process/steps for updating the TLS protocol and ciphers varies for different tools and libraries. **Following are few examples**:

For Browsers:

Ensure you are using latest version of browser which supports strong TLS 1.2 ciphers and TLS 1.3.

For JAVA client:

Please refer Configure Oracle's JDK and JRE Cryptographic Algorithms (java.com) on how to "Improve TLS cipher suite order"