Security Risk Report

The OpenSSL Software Foundation has published CVE-2021-3450 through a security bulletin on March 18, 2021. See https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-3450 for details.

This vulnerability bypasses additional security checks of certificates within a certificate chain, when the flag X509_V_FLAG_X509_STRICT was enabled, but no additional "purpose" was set by the application. This flag is enabled in our Unified Automation C++ and C based OPC UA SDKs. An already trusted leaf certificate may be used as if it was a CA (to create chains by signing other certificates). If the application additionally allows missing CRL (certificate revocation list), this may lead to authenticated secured access to the OPC UA application and may also lead to user rights escalation (in case of X509 user certificates are used).

Rating:

The vulnerability is rated with a Base Score of 6.6 and a Temporal Score of 6.1 according to the CVSS v3.1 metric with the following vector:

AV:N/AC:H/PR:H/UI:N/S:U/C:H/I:H/A:H/E:F/RL:O/RC:C

Details regarding prevention:

The vulnerability only affects C++ and C based OPC UA Clients and OPC UA Servers that use affected versions of the OpenSSL (affected v1.1.1h to v1.1.1j), in combination with allowing missing CRLs. If the application is configured to ignore revocation list, an upgrade to OpenSSL 1.1.1k is required. Being affected by the vulnerability in OpenSSL is prevented, when presence of CRLs is enforced (default). When using self-signed certificates, applications should never ignore/suppress CRL errors.

Affected Products/Versions:

OpenSSL (third-party) is not part of the SDK delivery, except for evaluation editions and examples. Customers are responsible for choosing the latest OpenSSL version. All C++ and C based OPC UA SDK products, which use the affected OpenSSL versions, are affected.

- **C++ based OPC UA Client Server SDK**
  (C++ SDK v1.7.4 contains example with affected OpenSSL v1.1.1j)
- **ANSI C based OPC UA Client Server SDK**
  (never contained affected OpenSSL, but you may have upgraded to it)
- **High Performance OPC UA SDK**
  (never contained affected OpenSSL, but you may have upgraded to it)

Important: If you are using the OPC UA hybrid protocol binding (UA Binary over HTTPS), you most likely use TLSv1.2. In that case you should immediately disable the https-endpoints, and upgrade to OpenSSL 1.1.1k. Within our Unified Automation SDK, the hybrid binding is marked experimental, is switched off by default, and should not be used in production.

Recommendation:

- Independent of UA SDK and OpenSSL version, and independent on the usage of self-signed or CA-signed certificates, make sure your application does not allow missing CRL (certificate revocation list). The application's configuration for ignoring CRL errors should be switched off/false (default).
- If your application uses C++ based OPC UA SDK, or AnsiC based OPC UA SDK, or High Performance SDK in combination with one of the three affected OpenSSL versions (v1.1.1h, v1.1.1i, v1.1.1j) you should upgrade to the latest OpenSSL v1.1.1k.
- If your application uses the "experimental" OPC UA hybrid protocol binding (UA-Binary over HTTPS) you should disable the https-endpoints and upgrade to the latest OpenSSL v1.1.1k.

FAQ:

- **Are any runtime products affected, like the UaGateway or the UaExpert?** No, the UaGateway and UaExpeert were shipped with OpenSSL version which is not affected by this vulnerability. However, the Linux edition may be affected, because it uses the system wide OpenSSL version.