The Article 24 (4) of eIDAS regulation.

The eIDAS Regulation (EU) No 910/2014 **http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:32014R0910**

Article 24(2) point (k): in case of qualified trust service providers issuing qualified certificates, establish and keep updated a **certificate database**.

Article 24(3): If a qualified trust service provider issuing qualified certificates decides to revoke a certificate, it **shall register** such **revocation** in **its certificate database** and publish the revocation status of the certificate in a timely manner, and in any event **within 24 hours** after the receipt of the request. The revocation shall become effective immediately upon its publication.

Article 24(4): With regard to paragraph 3, qualified trust service providers issuing qualified certificates shall provide to any relying party information on the **validity** or **revocation** status of qualified certificates issued by them. This information shall be made available at least **on a per certificate basis at any time and beyond the validity period** of the certificate in an automated manner that is reliable, free of charge and efficient.

The OCSP response is already able to cover the requirements like "beyond the validity period of the certificate" with the one of the two already used OCSP response extensions in the real systems:

- *CertHash* OCSP single extension (see **http://www.common-pki.org/uploads/media/Common-PKI_v2.0.pdf**),
- *ArchiveCutoff* OCSP extension (see RFC 6960 **https://tools.ietf.org/html/rfc6960#section-4.4.4**).

The *CertHash* - OCSP single extension is mandatory in some EU member states.

OCSP - *CertHash* is the hash value of the certificate whose status is returned by the OCSP response (Common PKI extensions *CertHash* (positive statement), Clause 3.1.2, Common PKI Specification V2.0). If this extension is found in the OCSP response, then the certificate **status is known** for OCSP and the hash value ensures the **integrity** by currently secure hash algorithm.

ITU-T/ISO standard for X.509 certificates and RFC 6960 define the rules for CRL and OCSP response.
RFC 6960:
"

thisUpdate - The **most recent time** at which the status being indicated **is known** by the responder **to have been correct**.
nextUpdate - The time at or before which newer information will be available about the status of the certificate.
producedAt - The time **at which** the OCSP responder **signed** this **response**.
revocationTime - The time at which the **certificate was revoked** or placed on hold.

If *nextUpdate* is not set, the responder is indicating that newer revocation information is available all the time.
"

Instead of the usage of the time value of OCSP *thisUpdate* field, some standards incorrectly use the value of OCSP field ***producedAt*** to determine the certificate status.
An example of "**ISO/DIS 32000-2.3**" in Chapter 12.8.3.4.8 "Revocation checking CAdES signatures at a time in the past" requires inappropriate comparison of the time value of OCSP field ***producedAt*** with the time-stamp time value. The correct comparison of the time-stamp time value must be with the time value of OCSP *thisUpdate* field and the value of *thisUpdate* field must be after the time-stamp time value.

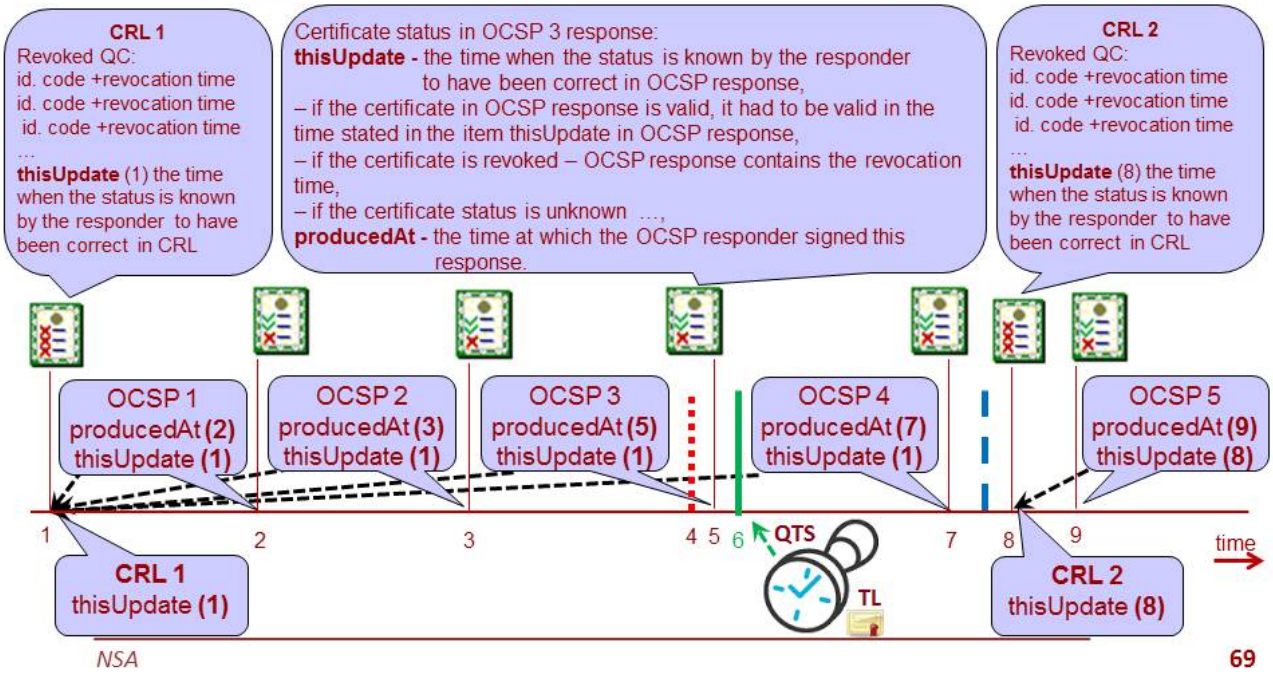
The example with incorrect result is illustrated in the following picture where **the certificate that is revoked** before the time-stamp time is declared incorrectly **as being valid** using OCSP 4 issued (***producedAt***) after the time-stamp time.

- In the time (6) (green line) the signature time-stamp is issued.
- Revocation of the certificate will occur in the time (4) (red dashed line).
- OCSP is based on data from CRL. For OCSP service there are available regular CRLs (in this case CRL 1 and CRL 2), whereas the OCSP service does not know about the issuance of an irregular CRL (issued at the certificate revocation) and if it does it is only accidentally when it checks the distribution http address for obtaining CRL (it is an inefficient method).
- The certificate was revoked in the time (4) when the irregular CRL was issued. The irregular CRL contains the time (4) in the CRL *revocationDate* field and the time (4) in the CRL *thisUpdate* field.
- OCSP 4 that is issued in the time (7) contains the time value (7) in the OCSP *producedAt* field. Since the issuance time of the revocation information (7) is after the time from the time-stamp (6), this revocation information must be used according to vague ETSI TS/EN 319 102-1 rules as OCSP response **with the final status**. This is a serious mistake because OCSP 4 contains old information being updated in the time (1) as indicated in the item OCSP *thisUpdate* field containing the time (1).

Information on the certificate revocation is provided only in **OCSP 5** that contains correct information updated according to data from CRL 2, thus OCSP 5 contains the update time (8) in the OCSP ***thisUpdate*** field and the OCSP ***revocationTime*** **field contains the revocation time (4).**

THE UPDATE TIME OF CRL OR OCSP RESPONSE USED FOR THE VALIDATION OF THE QC STATUS

Article 24 (4) of Regulation (EU) No 910/2014 – information on validity status or QC revocation



Certificate validation

Table 1 - CRL

1.		**if** (certificate.`notBefore` < *CRL*.`thisUpdate`) **and** (((CRL.`expiredCertsOnCRL` <= certificate.`notAfter`) **and** (0 < CRL.`expiredCertsOnCRL`)) **or** ((*CRL*.`thisUpdate` <= certificate.`notAfter`) **and** (0 = CRL.`expiredCertsOnCRL`))) **then**
2.		**if** certificate **is not revoked in** *CRL* **then**
3.		**if** *control-time* <= *CRL*.`thisUpdate` **then**
		VALID
4.		**else**
		WAS VALID at [*CRL*.`thisUpdate`**], the later status is not confirmed.** **If you need a confirmation of the later status, try to get a newer updated CRL.** **INDETERMINATE**
5.		**else**
		if *control-time* < *CRL*[certificate].`revocationDate` **then**
		VALID
6.		**else**
		INVALID – revoked at [*CRL*[certificate].`revocationDate` **]**
7.		**else**
		INDETERMINATE (INCOMPLETE AUTOMATIC VERIFICATION: a request to CA for CRL that can contain the status of the certificate being verified.)

With the meaning:

If CRL.`expiredCertsOnCRL` is not present in the CRL extension, then its value is 0, otherwise the value is defined according to ITU-T X.509.

CRL.`thisUpdate` is the time when the certificate status was updated, what means the certificate status will not be changed to `revoked` with the time value before the `thisUpdate` time in any time later.

Certificate.`notBefore` is the time since when it is possible to use the certificate and its status can be included in CRL.

Certificate.`notAfter` is the time after which the certificate status in CRL is not changed anymore but the status can be included in CRL.

CRL[certificate].`revocationDate` is the date of the certificate revocation being included in CRL.

Table 2 – OCSP response

1. **if** (certificate.`notBefore` < *OCSP*[certificate].`thisUpdate`) **and**
   (((OCSP.`ArchiveCutoff` <= certificate.`notAfter`) **and** (0 < OCSP.`ArchiveCutoff`)) **or**
   ((*OCSP*[certificate].`thisUpdate` <= certificate.`notAfter`) **and** (0 = OCSP.`ArchiveCutoff`)) **or**
   (*OCSP*[certificate].`CertHash` = certificate.*CertHash)*) **then**
2. **if** *OCSP*[certificate].`CertStatus` = `good` **then**
3. **if** *control-time* <= *OCSP*[certificate].`thisUpdate` **then**
   VALID
4. **else**
   WAS VALID at [*OCSP*[certificate].`thisUpdate`**], the later status is not confirmed.**
   If you need a confirmation of the later status, try to get a newer updated OCSP response.
   INDETERMINATE
5. **else**
   if *OCSP*[certificate].`CertStatus` = `revoked` **then**
   if *control-time* < *OCSP*[certificate].`revocationTime` **then**
   VALID
6. **else**
   INVALID - revoked at [*OCSP*[certificate].`revocationTime` **]**
7. **else**
   INDETERMINATE (INCOMPLETE AUTOMATIC VERIFICATION:
   OCSP does not know the current status of the certificate validity because
   OCSP[certificate].CertStatus = unknown Verification is possible by other OCSP or CRL.)
8. **else**
   INDETERMINATE (INCOMPLETE AUTOMATIC VERIFICATION: a request to CA
   for OCSP response or CRL that can contain the status of the certificate
   being verified.)

With the meaning:

OCSP.`ArchiveCutoff` - if `ArchiveCutoff` is not present in the OCSP response, then its value is 0, otherwise the value stored in `ArchiveCutoff` is defined according to RFC 6960 X.509 Internet Public Key Infrastructure, Online Certificate Status Protocol - OCSP.

OCSP[certificate].`CertHash` is the hash value of the certificate whose status is returned by the OCSP response (Common PKI extensions CertHash (positive statement), Clause 3.1.2, Common PKI Specification V2.0 www.common-pki.org). If this extension is found in the OCSP response, then the certificate status is known for OCSP and the hash value ensures the integrity by currently secure hash algorithm.

Certificate.`CertHash` is the hash value of the certificate whose status is verified.

OCSP.`producedAt` is the time of the OCSP response issuance.

OCSP[certificate].`thisUpdate` is the time when the certificate status was updated, what means the certificate status will not be changed to `revoked` with the time value before the thisUpdate time in any time later. The value must be smaller or equal to OCSP.`producedAt`.

OCSP[certificate].`nextUpdate` is the auxiliary time about the availability of the latest occurrence of the information about the status. The OCSP response must not contain the item `nextUpdate` if the certificate, whose status is returned, is expired.

Certificate.`notBefore` is the time since when it is possible to use the certificate and the certificate status can be included in OCSP (CRL).

Certificate.`notAfter` is the time after which the certificate status in CRL (OCSP) cannot be changed but the certificate status can be included in CRL (OCSP).

OCSP[certificate].`revocationTime` is the time of the certificate revocation.

OCSP[certificate].`CertStatus` is the status of the certificate being verified with the values: `good`, `revoked` and `unknown`.