Test specification Revocation-B2A (for both tesexecutions Android and IOS!)

TC-ID	Testcase	Description	Manual test steps			
TXR-5530	Revok_B2A_VerifierApp_VAC_Revoked	Given two certificates of type VAC -- *Certificate A* and *Certificate B*, both of which have NOT been revoked are to be scanned by the verifier app. Check that the verifier app determines that the scanned certificates are valid. A Certificate of type VAC -- *Certificate C* , which has been revoked is to be scanned by the verifier app. Check that the verifier app determines that the scanned Certificate is listed in the revocation list, i.e. it has been revoked.		Step	Input/Data	Expected Results
			1	Update the revocation lists on Verifier App. Scan both certificates A and B with the verifier app.	Certificate A: valid, not revoked Certificate B: valid, not revoked Certificate C: valid, but revoked	After scanning, the verifier app determines both certificates as valid.
			2	A revoked VAC-Certificate **Certificate C** is presented for scanning to the verifier app.	The same QR-Code as in TXR-5549, Revok_B2A_WalletApp_VAC_Revoked (to ensure, that Wallet- and Verifier App should bring the same result of the validation!)	After scanning, the verifier app determines the VAC-Certificate as revoked.
TXR-5543	Revok_B2A_VerifierApp_TEST_Revoked	Given two certificates of type TEST -- *Certificate A* and *Certificate B*, both of which have NOT been revoked are to be scanned by the verifier app. Check that the verifier app determines that the scanned certificates are valid. A Certificate of type TEST -- *Certificate C*, which has been revoked is to be scanned by the verifier app. Check that the verifier app determines that the scanned Certificate is listed in the revocation list, i.e. it has been revoked.		Step	Input/Data	Expected Results
			1	Update the revocation lists on Verifier App. Scan both TEST-Certificates A and B with the verifier app.	Certificate A: valid, not revoked Certificate B: valid, not revoked Certificate C: valid, but revoked	After scanning, the verifier app determines both certificates as valid.
			2	A revoked TEST-Certificate, **Certificate C**, is presented for scanning to the verifier app.	The same QR-Code as in TXR-5553 for Wallet App: (to ensure, that Wallet- and Verifier App should bring the same result of the validation!)	After scanning, the verifier app determines the TEST-Certificate as revoked.
TXR-5544	Revok_B2A_VerifierApp_REC_Revoked	Given two certificates of type REC -- *Certificate A* and *Certificate B*, both of which have NOT been revoked are to be scanned by the verifier app. Check that the verifier app determines that the scanned certificates are valid. A Certificate of type REC -- *Certificate C*, which has been revoked is to be scanned by the verifier app. Check that the verifier app determines that the scanned Certificate is listed in the revocation list, i.e. it has been revoked.		Step	Input/Data	Expected Results
			1	Update the revocation lists on Verifier App. Scan both REC-Certificates A and B with the verifier app.	Certificate A: valid, not revoked Certificate B: valid, not revoked Certificate C: valid, but revoked	After scanning, the verifier app determines both certificates as valid.
			2	A revoked REC-Certificate, **Certificate C**, is presented for scanning to the verifier app.	The same QR-Code as in TXR-5551 for Wallet App; (to ensure, that Wallet- and Verifier App should bring the same result of the validation!).	After scanning, the verifier app determines the REC-Certificate as revoked.
TXR-5546	Revok_B2A_VerifierApp_VAC_Revoked_Expired	A Certificate of type VAC which has been revoked at some point but the revocation has expired. The thus valid VAC is to be scanned by the verifier app. This test checks whether the VAC certificate is determined as valid by the verifier app.		Step	Input/Data	Expected Results
			1	A VAC-Certificate which has at some point been revoked but the revocation has expired, is presented for scanning to the verifier app.	The same QR-Code as TXR-5552. (to ensure, that Wallet- and Verifier App should bring the same result of the validation!)	After scanning, the verifier app determines the VAC-Certificate as valid.

TXR-5547	Revok_B2A_VerifierApp_TEST_Revoked_Expired	A Certificate of type TEST which has been revoked at some point but the revocation has expired. The thus valid TEST is to be scanned by the verifier app. This test checks whether the VAC certificate is determined as valid by the verifier app.		Step	Input/Data	Expected Results
			1	A TEST-Certificate which has at some point been revoked but the revocation has expired, is presented for scanning to the verifier app.	The same QR Code as in TXR-5553. (to ensure, that Wallet- and Verifier App should bring the same result of the validation!)	After scanning, the verifier app determines the TEST-Certificate as valid.
TXR-5548	Revok_B2A_VerifierApp_REC_Revoked_Expired	A Certificate of type REC which has been revoked at some point but the revocation has expired. The thus valid REC is to be scanned by the verifier app. This test checks whether the REC certificate is determined as valid by the verifier app.		Step	Input/Data	Expected Results
			1	A REC-Certificate which has at some point been revoked but the revocation has expired, is presented for scanning to the verifier app.	The same QR Code as in TXR-5554. (to ensure, that Wallet- and Verifier App should bring the same result of the validation!)	After scanning, the verifier app determines the REC-Certificate as valid.
TXR-5549	Revok_B2A_WalletApp_VAC_Revoked	Given two certificates of type VAC -- *Certificate A* and *Certificate B*, both of which have NOT been revoked are to be loaded by the wallet app. Check that the wallet app determines that the scanned certificates are valid. A revoked Certificate of type VAC -- *Certificate C*, which has been is saved in the wallet app. Check that the wallet app determines that the Certificate is listed in the revocation list, i.e. it has been revoked.		Step	Input/Data	Expected Results
			1	**Certificate A** of type VAC is **NOT** revoked. **Certificate B** of type VAC is **NOT** revoked Load both certificates in the wallet app.	Certificate A: valid, not revoked Certificate B: valid, not revoked Certificate C: revoked	After loading, the wallet app determines both certificates as valid.
			2	A revoked VAC-Certificate, **Certificate C**, is saved in the wallet app. The certificate is loaded into memory and checked for validation within the wallet app.	The same QR-Code as in TXR-5530, Revok_B2A_VerifiertApp_VAC_Revoked; (to ensure, that Wallet- and Verifier App should bring the same result of the validation!).	The wallet app evaluates the certificate as revoked.
TXR-5550	Revok_B2A_WalletApp_TEST_Revoked	A revoked Certificate of type TEST, *Certificate C*, which has been is saved in the wallet app. Check that the wallet app determines that the Certificate is listed in the revocation list, i.e. it has been revoked. Given two certificates of type TEST -- *Certificate A* and *Certificate B*, both of which have NOT been revoked are to be loaded in the wallet app. Check that the wallet app determines that the loaded certificates are valid.		Step	Input/Data	Expected Results
			1	**Certificate A** of type TEST is **NOT** revoked. **Certificate B** of type TEST is **NOT** revoked Load both certificates in the wallet app.	Certificate A: valid, not revoked Certificate B: valid, not revoked Certificate C: revoked	After loading, the wallet app determines both certificates as valid.
			2	A revoked TEST-Certificate is saved in the wallet app. The certificate is loaded into memory and checked for validation within the wallet app.	The same QR-Code as in TXR-5543, Revok_B2A_VerifiertApp_TEST_Revoked; (to ensure, that Wallet- and Verifier App should bring the same result of the validation!).	The wallet app evaluates the certificate as revoked.
TXR-5551		Given two certificates of type REC -- *Certificate A* and *Certificate B*, both of which have NOT been revoked are to be loaded by the wallet app. Check that the wallet app determines that the scanned certificates are valid.		Step	Input/Data	Expected Results
			1	**Certificate A** of type REC is **NOT** revoked.	Certificate A: valid, not revoked	

	Revok_B2A_WalletApp_REC_Revoked	A revoked Certificate of type REC -- *Certificate C*, which has been is saved in the wallet app. Check that the wallet app determines that the Certificate is listed in the revocation list, i.e. it has been revoked.		**Certificate B** of type REC is **NOT** revoked Load both certificates with the wallet app.	Certificate B: valid, not revoked Certificate C: revoked	After loading, the wallet app determines both certificates as valid.
			2	A revoked REC-Certificate is saved in the wallet app. The certificate is loaded into memory and checked for validation within the wallet app.	The same QR-Code as in TXR-5544, Revok_B2A_VerifiertApp_REC_Revoked; (to ensure, that Wallet- and Verifier App should bring the same result of the validation!).	The wallet app evaluates the certificate as revoked.
TXR-5552	Revok_B2A_WalletApp_VAC_Revoked_Expired	A Certificate of type VAC which has been is saved in the wallet app has been revoked but the revocation has expireed. Check that the wallet app determines the Certificate as valid.		Step	Input/Data	Expected Results
			1	A revoked VAC-Certificate whose revocation has expired, is saved in the wallet app. The certificate is loaded into memory and checked for validation within the wallet app.	The same QR Code as in TXR-5546 (to ensure, that Wallet- and Verifier App should bring the same result of the validation!)	The wallet app evaluates the certificate as valid.
TXR-5553	Revok_B2A_WalletApp_TEST_Revoked_Expired	A Certificate of type TEST which has been is saved in the wallet app has been revoked but the revocation has expired. Check that the wallet app determines the Certificate as valid.		Step	Input/Data	Expected Results
			1	A revoked TEST-Certificate whose revocation has expired, is saved in the wallet app. The certificate is loaded into memory and checked for validation within the wallet app.	The same QR-Code as in TXR-5547. (to ensure, that Wallet- and Verifier App should bring the same result of the validation!)	The wallet app evaluates the certificate as valid.
TXR-5554	Revok_B2A_WalletApp_REC_Revoked_Expired	A Certificate of type REC which has been is saved in the wallet app has been revoked but the revocation has expired. Check that the wallet app determines the Certificate as valid.		Step	Input/Data	Expected Results
			1	A revoked REC-Certificate whose revocation has expired, is saved in the wallet app. The certificate is loaded into memory and checked for validation within the wallet app.	The same QR Code as in TXR-5548. (to ensure, that Wallet- and Verifier App should bring the same result of the validation!)	The wallet app evaluates the certificate as valid.
TXR-5629	Revok_B2A_WalletApp_Config_Autom_Revocation_Update_WF_VAC_REC_TEST	- Test issue is here the automatic Update-Workflow-Process for all certificates claimed in Wallet App;		Step	Input/Data	Expected Results
			1	All four certificates have been claimed in the wallet app Then revoke the certificates A, B and C via GW-API.	Time is within the next configured X-hour cycle before the next automatic revocation update cycle is run on the wallet app; Certificate VAC A, TEST B and REC C: marked as revoked on GW but not updated on the National Backend;	The wallet app evaluates all three certificates as valid (passive evaluation).
			2	Trigger or wait for the Download of the revocation list by National backend.		Download is completed and the revocations are updated on the National Backend.
			3	At the configured time X, the automatic revocation check on the wallet app gets triggered and runs completely.		The wallet app evaluates automatically all three certificates as revoked. The three certificates are greyed out and marked with a red border and a hint "Certificate was invalidated by the issuer"
TXR-5655		- Test issue is here the configured Delete-Workflow-Process on VerifierApp (for e.g. VAC and REC)		Step	Input/Data	Expected Results

	Revok_B2A_VerifierApp_Config_Delete-WF_on_Verifier_VAC_REC	- Given two certificates of type VAC -- *Certificate A* and *Certificate B* , and two certificates of type REC -- *Certificate C* and +Certificate D*, all four of which have been revoked. Yet the revocation entries for *Certificate B* and *Certificate D* have expired. Thereupon, the Delete Workflow for the Verifier has run and the revocation entries for *Certificate B* and *Certificate D* have been deleted locally on the Verifier. Check that upon scanning all certificates, the verifier evaluates *Certificate B* and *Certificate D* as valid and *Certificate A* and *Certificate C* as revoked.	1	Scan all certificates with the verifier app.	The Delete-WF is configured to start at x o'clock; Current time is 5 Minutes to x o'clock; Certificate A: revoked; Certificate B: revoked; Certificate C: revoked; Certificate D: revoked.	Certificate A is evaluated as invalid (revoked); Certificate B is evaluated as revoked. Certificate C is evaluated as revoked; Certificate D is evaluated as revoked.
			2	Wait until the configured time x hr + 5 Minutes.		
			3	Scan the certificates again.	Current time is 5 Minutes after the configured time for the Delete Work Flow to run.	Certificate A is evaluated as invalid (revoked); Certificate B is evaluated as **valid** because the expired revocation entry has been deleted. Certificate C is evaluated as revoked; Certificate D is evaluated as **valid** because the expired revocation entry has been deleted.
TXR-5658	Revok_B2A_VerifierApp_Config_Download_WF_from_NB_VAC_REC	- Test issue is here the configuration and the Update-Workflow-Process of the revocation-list (means download from National Backend to Verifier) and the consistence of the Delete/Remove-Workflow of Verifier App and Nationa Backend. - Given two certificates of type VAC -- *Certificate A* and *Certificate B*, and two more of type REC -- *Certificate C* and *Certificate D*, all four of which have been revoked. Yet the revocation entries for *Certificate B* and *Certificate D* have expired. Thereupon, the Delete Workflow for the Verifier has run and the revocation entries for *Certificate B* and *Certificate D* have been deleted locally on the Verifier. Additionally, a sync download from the National Backend has been done. Check that upon scanning all four certificates, the verifier evaluates *Certificate B* and *Certificate D* as still valid (and not revoked) and *Certificate A* and *Certificate C* as still revoked.		Step	Input/Data	Expected Results
			1	Ensure the timepoint for the daily start of the Delete-WF on the Verifier App is configured at 8:00 am and the Download-timepoint from NB at 12:00 am.		
			2	Scan all certificates with the verifier app after the Delete-Workflow on the Verifier and before sync Download from NB.	5 Minutes before sync Download from NB (National Backend) Certificate A: revoked; Certificate B: valid (revocation expired); Certifcate C: revoked; Certificate D: valid (revocation expired.	Certificate A is evaluated as invalid (revoked); Certificate B is evaluated as **valid** because the revocation entry has been deleted. Certificate C is evaluated as revoked;

						Certificate D is evaluated as **valid** because the revocation entry has been deleted.
			3	Trigger sync Download from National Backend as configured.		Download is carried out and completed without error.
			4	Wait for sync Download from NB (National Backend) to complete. Scan all certificates with the verifier app.	5 Minutes after sync Download from NB has been completed.	The result of the validation is the same as before the Download from the NB on VerifierApp: Certificate A is evaluated as invalid (revoked); Certificate B is evaluated as **valid**; Certificate C is evaluated as revoked; Certificate D is evaluated as **valid**.
TXR-5669	Revok_B2A_VerifierApp_KID_With_More_Than_1000_Revoked_Entries	- Check that a validation of a revoked Certificate also works for a revocation list of a KID with more than 1000 revoked entries.		Step	Input/Data	Expected Results
			1	Ensure that there are more than 1000 revoked entries of the same KID and the same expired date are uploaded on the GW (means "post" at least two batches with the same KID and Expired data on GW, each of which contains one of the two revoked certificate A or B).		
			2	Scan both certificates A and B from the same KID with the verifier app.	Certificate A: valid and revoked in batch_A Certificate B: valid and revoked in batch_B	After scanning, the verifier app determines: Certificate A: revoked Certificate B: revoked
TXR-5670	Revok_B2A_WalletApp_KID_With_More_Than_1000_Revoked_Entries	- Check that a validation in the WalletApp of a revoked Certificate also works for a revocation list of a KID with more than 1000 revoked entries.		Step	Input/Data	Expected Results
			1	Ensure that there are more than 1000 revoked entries of the same KID and the same expired date are uploaded on the GW (means "post" at least two batches with the same KID and Expired data on GW, which contain certificate A and B).		
			2	Choose certificate A in WalletApp and start the validation for this certificate A.	Certificate A: valid, not revoked Certificate B: valid but revoked	The Wallet App shows that the certificate A is valid and not revoked
			3	Choose certificate B in WalletApp and start the validation for this certificate B.		The Wallet App shows, that the certificate B is revoked.
TXR-5701	Revok_B2A_VerifierApp_Delete_Flag_Set_By	Test issue here is the automatic deletion of entries from batches where the "Delete" Flag has been set by the member state, although the revocation entries have not expired (Expiry Date is still in the future);		Step	Input/Data	Expected Results
			1	Given two certificates of type VAC – **Certificate A** and **Certificate B** , and two certificates of type REC – **Certificate C** and **Certificate D**, all four of which are revoked and the revocation entries have not expired (Expiry Date is still in the future). Scan all certificates with the verifier app.	Certificate A: revoked, not expired; Certificate B: revoked, not expired; Certificate C: revoked, not expired;	Certificate A is evaluated as revoked; Certificate B is evaluated as revoked.

	Revok_B2A_VerifierApp_Delete_Flag_Set_By_MS_VAC_REC				Certificate D: revoked, not expired.	Certificate C is evaluated as revoked; Certificate D is evaluated as revoked.
			2	Member state decides to flag the batch for deletion, although the expiry date is still in the future. The **DELETE**-Flag gets set to TRUE.		
			3	After the verifier has synchronised with the NB, the four certificates are scanned a second time.		Certificate A, B, C and D are evaluated as valid (not revoked) because the revocation batch is deleted.
TXR-5737	Revok_B2A_VerifierApp_Validation_After_Batch_Delete_By_MS	The indirect test issue here is the actively observation by the GW to remove the batches which are directly requested to delete by a Member State. The check of this ant its impact will be indirectly verified by the validation of a QR-Code in a batch, which should be deleted by the GW before.		Step	Input/Data	Expected Results
			1	Given two certificates of type VAC – **Certificate A** and **Certificate B** , both of which are revoked and the revocation entries have not expired. Scan the certificates with the verifier app.	Batch with Certificate A (revoked, not expired); Batch with Certificate B: revoked, not expired;	Certificates A and B are evaluated as revoked;
			2	Member State deletes the batch containing the revocation entry B, using the GW-Delete-API for the batch.		API-Call is caried out with no error.
			3	After the download of the revocation lists from the GW by NB and the verifier app has then synchronised with the NB, the two certificates A and B are scanned a second time again.		Certificate A is evaluated as revoked; Certificate B is evaluated as valid (not revoked) because the batch with the revocation entry is deleted by the Member State.
TXR-5857	Revok_B2A_Verifier_KID_UNKNOWN_VAC_REC	- Test issue is here is the upload of revocation entries (for e.g. VAC and REC) in a List with an "UNKNOWN KID" (The KID value is UNKNOWN). - Given two certificates -- *Certificate VAC_C1* and *Certificate REC_C1* , which have been revoked and the revocation entries have been uploaded to a list with an UNKNOWN KID and this list was already distributed over the Gateway. Check whether the certificates -- *Certificate VAC_C1* and *Certificate REC_C1* -- are evaluated as revoked by the verifier app.		Step	Input/Data	Expected Results
			1	Two certificates – **Certificate VAC_C1** and **Certificate REC_C1**, have been revoked and the revocation entries have been uploaded to a list with an UNKNOWN KID and this list was already distributed over the Gateway.	VAC_C1: revoked, entry not expired; REC_C1: revoked, entry not expired; KID = 'UNKNOWN'	
			2	Scan all certificates with the verifier app.		Certificate VAC_C1 is evaluated as revoked; Certificate REC_C1 is evaluated as revoked.
TXR-5871		- Test issue is here is the upload of revocation entries (for e.g. VAC and REC) in a List with an "UNKNOWN KID" (The KID value is UNKNOWN).		Step	Input/Data	Expected Results

	Revok_B2A_Wallet_KID_UNKNOWN_VAC_REC	- Given two certificates -- *Certificate VAC_C1* and *Certificate REC_C1* , which have been revoked and the revocation entries have been uploaded to a list with an UNKNOWN KID and this list was already distributed over the Gateway. Check whether the certificates -- *Certificate VAC_C1* and *Certificate REC_C1* -- are evaluated as revoked by the wallet app.	1	Two certificates – **Certificate VAC_C1** and **Certificate REC_C1**, have been revoked and the revocation entries have been uploaded to a list with an UNKNOWN KID and this list was already distributed over the Gateway.	VAC_C1: revoked, entry not expired; REC_C1: revoked, entry not expired; KID = 'UNKNOWN'	
			2	Claim all certificates with the wallet app and load them vor evaluation.		Certificate VAC_C1 is evaluated as revoked; Certificate REC_C1 is evaluated as revoked.