DCMTK - Bug #1075

Use after free in dcmqrscp

2023-04-24 16:37 - Michael Onken

Status:	Closed	**Start date:**	2023-04-24
Priority:	Normal	**Due date:**	
Assignee:	Michael Onken	**% Done:**	0%
Category:	Application	**Estimated time:**	0:00 hour
Target version:			
Module:	dcmqrdb	**Compiler:**	
Operating System:			

Description

The storage (**) for the pointer () to the association object is located on the stack of waitForAssociation. When it calls handleAssociation(), the latter eventually calls destroyAssociation()**, which frees the resources and NULLs out the storage pointer passed to it, so as to prevent it from being reused.

However, handleAssociation never receives the original storage pointer (**). Instead, it uses the storage of its call arguments as they appear on the stack. destroyAssociation then overwrites handleAssociation's arguments, which are then discarded once it returns. Finally, waitForAssociation is not aware of the changes, since they did not modify its local storage pointer (which remains non-NULL), and it proceeds to call ASC_dropAssociation in clean-up, resulting in the UaF.

Thanks to Ahmad Hazimeh for the report and suggested patch.

History

#1 - 2023-04-24 16:38 - Michael Onken

- Status changed from New to Closed

Closed by commit 01ec789da5c2d88cdb77d7bf515a1a670c1f9638.