

# Uniface 9: **&#091;SOLVED&#093;** \$encode SHA256 with newline

## **&#091;SOLVED&#093;** \$encode SHA256 with newline

Author: kevin.mckeever@gmail.com (KevinMcKeever)

Hi Uniface sharp shooters, The following code correctly creates a SHA256 hash of a string:- message/info \$encode("HEX", \$encode("sha256", "abc123")) ;i.e 6CA13D52CA70C883E0F0BB101E425A89E8624DE51DB2D2392593AF6A84118090 ...however adding the new line character (hex 0A) i.e. uniface %%^ cannot be handled correctly with \$encode.....so message/info \$encode("HEX", \$encode("sha256", "abc%%^123")) ; = FB56F5FD48EF84D61B05A733FB5EB1252FAF4384EFB117B6FB9C7028BCFF83E8 when actually the correct result is ;= ab9c7f3dc9df904e527dbf7ce8da91d55115765069977afe23db4f111ac97821 (see an online calculator - e.g. <http://www.xorbin.com/tools/sha256-hash-calculator> or <http://onlinemd5.com/>) Is this a PBKAC (Problem between keyboard and chair) or a bug in \$encode? Also when SHA256 of empty string "" causes an error instead of creating the correct hash of e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855 Why?: I'm building out the Signature v4 signing process for calling an AWS API Gateway api which requires signing a canonical request with new line separators. BTW - MD5 with data containing %%^ also generates a different hash than the one online. Also tried using blockdata, but no joy.