NAME

au_bsm_to_errno, **au_errno_to_bsm**, **au_strerror** **-** convert between BSM and local error numbers

LIBRARY

Basic Security Module Library (libbsm, -lbsm)

SYNOPSIS

#include <bsm/libbsm.h>

int
au_bsm_to_errno(*u_char bsm_error*, *int *errorp*);

u_char
au_errno_to_bsm(*int error*);

*const char **
au_strerror(*int bsm_error*);

DESCRIPTION

These interfaces may be used to convert between the local (errno(2)) and BSM error number spaces found in BSM return tokens.

The **au_bsm_to_errno**() function accepts a BSM error value, *bsm_error*, and converts it to an errno(2) that will be stored in the integer pointed to by *errorp* if successful. This call will fail if the BSM error cannot be mapped into a local error number, which may occur if the return token was generated on another operating system.

The **au_errno_to_bsm**() function accepts a local errno(2) value, and returns the BSM error number for it. This call cannot fail, and instead returns a BSM error number indicating to a later decoder that the error could not be encoded.

The **au_strerror**() function converts a BSM error value to a string, generally by converting first to a local error number and using the local strerror(3) function, but will also work for errors that are not locally defined.

RETURN VALUES

On success, **au_bsm_to_errno**() returns 0 and a converted error value; on failure, it returns -1 but does not set errno(2).

On success, **au_strerror**() returns a pointer to an error string; on failure it will return NULL.

SEE ALSO

au_to_return(3), au_to_return32(3), au_to_return64(3), libbsm(3)

HISTORY

au_bsm_to_errno() and **au_errno_to_bsm**() were introduced in OpenBSM 1.1.

AUTHORS

These functions were implemented by Robert Watson under contract to Apple Inc.

The Basic Security Module (BSM) interface to audit records and audit event stream format were defined by Sun Microsystems.

BUGS

au_strerror is unable to provide localized strings for errors not available in the local operating system.