NAME

BUF_MEM_new, BUF_MEM_free, BUF_MEM_grow, BUF_strdup - simple character arrays structure

SYNOPSIS

```
#include <openssl/buffer.h>

BUF_MEM *BUF_MEM_new(void);

void BUF_MEM_free(BUF_MEM *a);

int BUF_MEM_grow(BUF_MEM *str, int len);

char * BUF_strdup(const char *str);
```

DESCRIPTION

The buffer library handles simple character arrays. Buffers are used for various purposes in the library, most notably memory BIOs.

The library uses the BUF_MEM structure defined in buffer.h:

```
typedef struct buf_mem_st
{
int length; /* current number of bytes */
char *data;
int max; /* size of buffer */
} BUF_MEM;
```

length is the current size of the buffer in bytes, **max** is the amount of memory allocated to the buffer. There are three functions which handle these and one "miscellaneous" function.

BUF_MEM_new() allocates a new buffer of zero size.

BUF_MEM_free() frees up an already existing buffer. The data is zeroed before freeing up in case the buffer contains sensitive data.

BUF_MEM_grow() changes the size of an already existing buffer to **len**. Any data already in the buffer is preserved if it increases in size.

BUF_strdup() copies a null terminated string into a block of allocated memory and returns a pointer to the allocated block. Unlike the standard C library *strdup()* this function uses *OPENSSL_malloc()* and so should be used in preference to the standard library *strdup()* because it can be used for memory leak checking or replacing the *malloc()* function.

The memory allocated from *BUF_strdup()* should be freed up using the *OPENSSL_free()* function.

RETURN VALUES

BUF_MEM_new() returns the buffer or NULL on error.

BUF_MEM_free() has no return value.

BUF_MEM_grow() returns zero on error or the new size (i.e. **len**).

SEE ALSO

bio(3)

HISTORY

BUF_MEM_new(), *BUF_MEM_free()* and *BUF_MEM_grow()* are available in all versions of SSLeay and OpenSSL. *BUF_strdup()* was added in SSLeay 0.8.