|  |  |
| --- | --- |
| Function | RESEARCH RESULTS |
| strcpy | Parameters:   * Pointer to char (used for the destination) * Pointer to constant char (used for the source)   Return Type:   * Pointer to char (the destination)   Example:  char one[] = {“This is one”};  char two[]= {“This is two”};  strcpy(one, two);  printf(“char one = %s”, one);  printf(“char two = %s”, two);  Function Bugs:   * If the destination string is smaller than the source, then something weird might happen * Overflow can be used to crack the machine and take control   (taken from the man page BUGS section)  Location of Prototype:   * string.h Prototype >> char \*strcpy(char \*dest, const char \*src); |
| strncat | Parameters:   * Pointer to char (used for the destination) * Pointer to constant char(used for the source) * size\_t n (unsigned integer used for the maximum bytes of the string)   Return Type:   * Pointer to the destination string   Example:  char one[24] = {“This is one”};  char two[12] = {“This is two”};  strncat(one, two, 11);  printf(“%s”, one);  Function Bugs:   * Possibility of data loss if size is too small   Taken from NOTES of the man page  Location of Prototype:   * string.h Prototype >> \*strncat(char \*dest, const char \*src, size\_t n); |
| strncasecmp | Parameters:   * Pointer to constant char (first string) * Pointer to constant char (second string) * size\_t n (unsigned integer used for the maximum bytes of the string)   Return Type:   * returns an int   Example:  char one[] = {“This is one”};  char two[] = {“This is two”};  int isItTheSame;  isItTheSame = strncasecmp(one, two, 11);  if(isItTheSame == 0)  printf(“They are the same”);  Function Bugs:   * I cant find any known bugs   Location of Prototype:   * strings.h Prototype >> int strncasecmp(const char \*s1, const char \*s2, size\_t n); |
| strstr | Parameters:   * Pointer to constant char (used for the larger string, or ‘haystack’) * Pointer to constant char (used for the smaller string, or ‘needle’)   Return Type:   * Pointer to char (smaller string found)   Example:  char one[] = {“This is one”};  char two[] = strstr(one, “This”);  printf(“%s”, two);  Function Bugs:   * It is case sensitive   Taken from DESCRIPTION of the man page  Location of Prototype:   * string.h Prototype >> char \*strstr(const char \*haystack, const char \*needle); |
| atoi | Parameters:   * Pointer to constant char   Return Type:   * Integer value of the string   Example:  char one[] = {“This is one”};  int var = atoi(one);  printf(“%d”, var);  Function Bugs:   * Does not detect errors * If the string value is higher than the max value of an int errors occur   Taken from DESCRIPTION of the man page  Location of Prototype:   * stdlib.h Prototype |
| fgets | Parameters:   * Pointer to char (used as a buffer) * Integer (used to specify the size of the string) * FILE structure (used to specify what stream you want to collect from)   Return Type:   * Pointer to char on success * NULL on failure   Example:  char one[12];  fgets(one, 12, stdin);  printf(“%s” , one);  Function Bugs:   * If a newline is read it is stored in the buffer   From DESCRIPTION of the man page  Location of Prototype:   * stdio.h Prototype >> char \*fgets(char \*s, int size, FILE \*stream); |