Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky, Katedra počítačov a informatiky

Inšpekcia kódu a sledovanie problémov

Lukáš Vavrek
B3
2015/2016

Obsah

1 Indentifikované chyby - cudzí projekt

1.1 Reverse calloc issue

file: bmp.c
method: char* reverse (const char*)
line: char* output = (char*) calloc (strlen(text)+1, sizeof(char*));

description: Nesprávne alokovanie pamäte. Char* má na 64b systémoch 8B a na 32b 4B.. Char ma pritom 1B
label: Únik pamäte

suggested fix: char* output = (char*) calloc (strlen(text)+1, sizeof(char));

1.2 Allocation check missing

file: bmp.c
method: char* reverse (const char*)
line: char* output = (char*) calloc (strlen(text)+1, sizeof(char*));

description: Po zavolaní funkcie calloc sa neskontroluje, či bola daná pamäť naozaj alokovaná a priradená. Nieje ošetrený prípad, kedy alokovaní zlyhá z dôvodu nedostatku pamäte.
label: Neošetrené vstupy

suggested fix: za volanie pridať if (! output) // ošetrenie

1.3 Missing string ending

file: bmp.c
method: char* reverse (const char*)
line: for(int j=0; j¡strlen(text); j++) output[j] = toupper(text[strlen(text)-1-j]);

description: Funkcia nahadzuje postupne písmena do "stringuöutput, no po skončení tohto foru neukončí output.
label: Prekročenie hranice polí

suggested fix: za for vložiť (j - musí byť správna pozícia) output [j] = '\0';

1.4 Double if

file: bmp.c
method: char* vigenere_encrypt(const char*, const char*)
line: if(key==NULL) return NULL; if(text==NULL) return NULL;

description: Zbytočne zdvojené 2 if-y za sebou
label: Nevhodne formátovaný kód

suggested fix: Spojiť ich do kopy pomocou ||

1.5 Variable name problem

file: bmp.c
method: char* vigenere_encrypt(const char*, const char*)
line: int tmp = 0;

description: tmp nieje vhodné meno premennej. vôbec to nenapovedá o tom na čo slúži
label: Zle pomenovanie

suggested fix: premenovať podľa významu (key_counter napr.)

1.6 Missing alocation check

file: bmp.c
method: char* vigenere_encrypt(const char*, const char*)
line: char* output = (char*) calloc (strlen(text)+1, sizeof(char));

description: po volani calloc nieje ošetrené, či sa alokovanie pamäti podarilo
label: Neošetrené vstupy

suggested fix: kontrolovať návratovú hodnotu funkcie

1.7 Long complicated line of logic

file: bmp.c
method: char* vigenere_encrypt(const char*, const char*)
line: output[j] = (toupper(text[j]) + toupper(key[tmp

description: nečítateľné
label: Nevhodne formátovaný kód

suggested fix: čitateľnosti napovie lepšie meno premennej tmp a vhodné rozdelenie na viac riadkov

1.8 Not using pre/post fix operators

file: bmp.c
method: char* vigenere_encrypt(const char*, const char*)
line: tmp=tmp+1;

description: nevyužívanie možnosti jazyka C
label: Nevhodne formátovaný kód

suggested fix: použiť prefixový alebo postfixový ++ operátor

2 Opravovanie chýb - vlastný projekt

2.1 Double usage of strlen

file: bmp.c
method: char* reverse(const char*)
line: char *reversed = (char *)malloc((strlen(text) + 1) * sizeof(char));
for (const char *ch = text + strlen(text) - 1; ch != text; ch–)

description: dvojité volanie funkcie strlen, pričom text sa nemení
label: Nevhodne formátovaný kód

suggested fix: pred prvým volaním uložiť hodnotu do premennej

---

comment: Opravene v 0cd39449 Pridane ulozenie hodnoty do premennej a nasledne jej vyuzivanie

gitdiff:

```
diff --git a/ps2/bmp.c b/ps2/bmp.c
index 909b49d..cd34db8 100644
--- a/ps2/bmp.c
+++ b/ps2/bmp.c
@@ -12,12 +12,14 @@ char* reverse(const char* text) {
  if (! text || empty_str (text))
  return NULL;

-char *reversed = (char *)malloc((strlen(text) + 1) * sizeof(char));
+ int text_length = strlen (text);
+
+ char *reversed = (char *)malloc((text_length + 1) * sizeof(char));
  if (! reversed)
  return NULL;
  char *r = reversed;

-for (const char *ch = text + strlen(text) - 1; ch != text; ch--) {
+ for (const char *ch = text + text_length - 1; ch != text; ch--) {
  *(r++) = toupper(*ch);
  }
```

2.2 Variable name problem

file: bmp.c
method: char* bit_encrypt(const char*)
line: for (const char *ch = text; *ch; ch++)

description: Nejasné pomenovanie premennej ch
label: Zlé pomenovanie

suggested fix: Premenovať na napr. text_ptr

---

comment: Fixed in f8061f8f Variable renamed to text_ptr

gitdiff:

```
diff --git a/ps2/bmp.c b/ps2/bmp.c
index cd34db8..28cd656 100644
--- a/ps2/bmp.c
+++ b/ps2/bmp.c
@@ -77,9 +77,9 @@ char* bit_encrypt(const char* text) {
  return NULL;
  char *e = encrypted;

-for (const char *ch = text; *ch; ch++) {
-unsigned char hnib = *ch & 240;
-unsigned char lnib = *ch & 15;
+ for (const char *text_ptr  = text; *text_ptr; text_ptr++) {
+ unsigned char hnib = *text_ptr & 240;
+ unsigned char lnib = *text_ptr & 15;

  hnib = ((hnib & 0b10101010) >> 1) | ((hnib & 0b01010101) << 1);
  lnib ^= (hnib >> 4);
```

2.3 Missing alocation check

file: playfair.c
method: char *playfair_cryptor(const char *, const char *, bool)
line: char *result = (char *)malloc((strlen(value) + 1) * sizeof(char));

description: Neošetrené alokovanie pamäti suggested
label: Neošetrené vstupy

suggested fix: Pridať kontrolu na NULL

---

comment: Fixed in 3a35caec Added NULL check

gitdiff:

```
diff --git a/ps2/playfair.c b/ps2/playfair.c
index 29c46e1..fa573fe 100644
--- a/ps2/playfair.c
+++ b/ps2/playfair.c
@@ -123,8 +123,15 @@ int get_index_from_point(const Point *p, int width) {

 char *playfair_cryptor(const char *key, const char *value, bool encrypting) {
   char *matrix_key = generate_matrix_key(key);
+ if (! matrix_key)
+ return NULL;

   char *result = (char *)malloc((strlen(value) + 1) * sizeof(char));
+ if (! result) {
+ free (matrix_key);
+ return NULL;
+ }
+
   char *r = result;

   short mult = encrypting ? 1 : -1;
@@ -185,4 +192,4 @@ char* playfair_decrypt(const char* key, const char* text) {
   return NULL;

   return playfair_cryptor(key, text, false);
-}
\ No newline at end of file
+}
```

2.4 Missing alocation check

file: playfair.c
method: char* generate_matrix_key(const char *)
line: char *matrix = (char *)malloc(26 * sizeof(char));

description: Neošetrené alokovanie pamäti suggested
label: Neošetrené vstupy

suggested fix: Pridať kontrolu na NULL

---

comment: Fixed in 3a35caec Added NULL check

gitdiff:

```
diff --git a/ps2/playfair.c b/ps2/playfair.c
index 29c46e1..fa573fe 100644
--- a/ps2/playfair.c
+++ b/ps2/playfair.c
@@ -123,8 +123,15 @@ int get_index_from_point(const Point *p, int width) {

 char *playfair_cryptor(const char *key, const char *value, bool encrypting) {
  char *matrix_key = generate_matrix_key(key);
+ if (! matrix_key)
+ return NULL;

  char *result = (char *)malloc((strlen(value) + 1) * sizeof(char));
+ if (! result) {
+ free (matrix_key);
+ return NULL;
+ }
+
  char *r = result;

  short mult = encrypting ? 1 : -1;
@@ -185,4 +192,4 @@ char* playfair_decrypt(const char* key, const char* text) {
  return NULL;

  return playfair_cryptor(key, text, false);
-}
\ No newline at end of file
+}
```

3 Záver

V kóde boli nájdene a opravené chyby rôzneho charakteru. Zmeny, ktoré boli vykonané na ich opravu, odstráňujú logické chyby v kóde a zlepšujú jeho čítateľnosť. Celkova kvalita kódu sa zvyšuje.