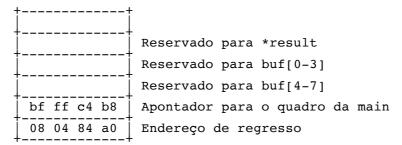
TPC9 e Guião Laboratorial

Dicas para a execução do trabalho

Este problema cobre uma gama variada de tópicos: quadros de funções na *stack* (*stack frames*), representação de *strings*, código ASCII, e ordenação de *bytes*. Este trabalho usa uma versão compilada sem otimização (para melhor compreender o processo de compilação), em que apenas o registo %ebp é salvaguardado antes de se reservar espaço para o vetor buf. Os valores de endereços indicados aqui poderão não corresponder aos que foram analisados nas aulas.

Há 2 pistas relevantes que deverão ser consideradas: (i) a sugestão dada no enunciado para testarem o código com uma *string* longa, e (ii) a indicação do compilador de que a função gets é "perigosa" (por ter "cadastro criminal"...). Assim, é de desconfiar que a anomalia poderá estar no uso do espaço de memória alocado à *string* (no quadro da função getline, na *stack*) e na utilização da função gets.

c) (A/R) Construção do quadro de getline na stack antes de chamar a função gets (stack cresce para cima): o endereço de regresso está no código da main (é o endereço da instrução a seguir a call), e o valor do apontador para o quadro da main obtém-se lendo o valor em %ebp após uma paragem no início de getline. Para confirmar, "examinar" em hexadecimal as 2 words (32 bits na terminologia do gdb) a partir da posição apontada por \$ebp.



d) (R) Estado do quadro de getline após regressar de gets (assumindo que foi introduzida a string 123456789012). A função gets limita-se a ler uma linha do standard input até encontrar o caráter newline ou uma condição de erro, terminando a seguir a escrita da string com o caráter null (ASCII 0x00); neste caso, vai ler os 12 carateres da string e acrescentar o null; mas como apenas tem reservado para a string um array de 8 elementos... veja-se o resultado (se se compilar com otimização -02, seria preciso introduzir bastantes mais valores para ter o mesmo efeito; quantos mais, e porquê?):

```
| 34 33 32 31 | buf[0-3] | | 38 37 36 35 | buf[4-7] | | 32 31 30 39 | Apontador para o quadro da main | 08 04 84 00 | Endereço de regresso
```

- e) (R) Este programa está a tentar regressar ao endereço 0x08048400, uma vez que o *byte* menos significativo **foi modificado** (*overwritten*) pelo caráter terminador (*null character*).
- f) (R) O valor guardado do apontador para o quadro da main foi modificado para 0x32313039, e este valor será o "recuperado" para %ebp antes do regresso de getline. O mesmo na versão compilada com -02, onde todos os registos salvaguardados seriam alterados.
- g) (B) A chamada de malloc deveria ter como argumento strlen(buf)+1, e deveria também verificar que o valor a devolver é non-null. Sugestão para evitar este tipo de problema: usar fgets ou scanf em vez de gets (que não é suportado desde 2011).