



Engenharia de Segurança
Grupo 7
Aula 10

Bruno Machado - A74941
Diogo Gomes - A73825
Francisco Mendes - A75097

Abril 2018

1

- No programa de java ocorre uma exceção de `ArrayIndexOutOfBoundsException` devido ao controlo de memória existente no java que não deixa que hajam escritas fora dos limites de um array.
- Tal como o java o Python também implementa controlo de memória impedindo que sejam feitas escritas fora dos limites do array, respondendo com uma mensagem de `IndexError`.
- Em C++ não existe controlo de memória, deste modo ao tentar escrever na posição 10, o programa vai escrever no espaço de memória seguinte, podendo ocorrer várias ações, entre elas o `segmentation fault`.

2

- No programa de java ocorre uma exceção de `ArrayOutOfBoundsException` devido ao controlo de memória existente no java que não deixa que hajam escritas fora dos limites de um array.
- Tal como o java o Python também implementa controlo de memória impedindo que sejam feitas escritas fora dos limites do array, respondendo com uma mensagem de `IndexError`.
- Em C++ não existe controlo de memória, como neste caso a variável que é preenchida antes do array é a variável `"i"`, o seu valor estará declarado acima do buffer. Se for pedido para guardar um número de caracteres maior que 10 que neste caso é o tamanho do buffer, o valor da variável `"i"` será alterada quando for escrito no índice 10 do array. Esta alteração pode levar a um loop infinito no caso de valores entre [11,19]. Valores superiores ou iguais a 20 produzem resultados imprevistos uma vez que reescrevem outra variáveis sem ser a variável `"i"`. Se o valor inserido for superior ao tamanho da stack do programa, leva a que ocorra o erro de `segmentation fault`.

3

Ambos os programas apresentam a vulnerabilidade conhecida como **Buffer Overflow**.

- **RootExploit:** ao exceder o tamanho reservado do buffer para receber o input da password pode "corromper" a `flag` que valida a password. Assim, apesar de se introduzir a palavra passe errada são dadas permissões de utilizador root.
- **0-simple:** à semelhança do programa anterior, ao exceder o tamanho do buffer que recebe o input pode-se alterar o valor da variável `control`, resultando na mensagem "YOU WIN!!!"

4

Neste programa, quando se declara que o tamanho da frase que se vai introduzir é relativamente grande, para além de imprimir lixo no final do buffer, foi possível aceder ao valor de algumas variáveis de ambiente:

```

.....
...../ReadOverflow.LS COL0
RS=rs=0;di=01;34;ln=01;36;mh=00;pi=40;33;so=01;35;do=01;35;bd=40;33;01;cd=40;33;01;or=40;31;01;mi=00;su=37;41;sg=30;43;ca=30;
41;tw=30;42;ow=34;42;st=37;44;ex=01;32;*tar=01;31;*tgz=01;31;*arc=01;31;*arj=01;31;*taz=01;31;*lha=01;31;*lza=01;31;*
lzh=01;31;*lzm=01;31;*tlz=01;31;*txz=01;31;*tzo=01;31;*t7z=01;31;*zip=01;31;*z=01;31;*Z=01;31;*dz=01;31;*gz=01;31;
*lrz=01;31;*lz=01;31;*lzo=01;31;*xz=01;31;*zst=01;31;*tztst=01;31;*bz2=01;31;*bz=01;31;*tbz=01;31;*tbz2=01;31;*tz=0
1;31;*deb=01;31;*rpm=01;31;*jar=01;31;*war=01;31;*ear=01;31;*sar=01;31;*rar=01;31;*alz=01;31;*ace=01;31;*zoo=01;31;
*.cpio=01;31;*7z=01;31;*rz=01;31;*cab=01;31;*jpg=01;35;*jpeg=01;35;*mjpg=01;35;*mjpeg=01;35;*gif=01;35;*bmp=01;35;*
pbm=01;35;*pgm=01;35;*ppm=01;35;*tga=01;35;*xbm=01;35;*xpm=01;35;*tif=01;35;*tiff=01;35;*png=01;35;*svg=01;35;*svgz
=01;35;*mng=01;35;*pcx=01;35;*mov=01;35;*mpg=01;35;*mpeg=01;35;*m2v=01;35;*mkv=01;35;*webm=01;35;*ogm=01;35;*mp4=01
;35;*m4v=01;35;*mp4v=01;35;*vob=01;35;*qt=01;35;*nuv=01;35;*wmv=01;35;*asf=01;35;*rm=01;35;*rmvb=01;35;*flc=01;35;*
.avi=01;35;*fli=01;35;*flv=01;35;*gl=01;35;*dl=01;35;*xcf=01;35;*xwd=01;35;*yuv=01;35;*cgm=01;35;*emf=01;35;*ogv=01
;35;*ogx=01;35;*aac=00;36;*au=00;36;*flac=00;36;*m4a=00;36;*mid=00;36;*midi=00;36;*mka=00;36;*mp3=00;36;*mpc=00;36;
*.ogg=00;36;*ra=00;36;*wav=00;36;*oga=00;36;*opus=00;36;*spx=00;36;*xspf=00;36;.LANG=en_US.UTF-8.GDM_LANG=en_US.utf8.DI
SPLAY=0.GTK_OVERLAY_SCROLLING=0.COLORTERM=truecolor.XDG_VTNR=7.SSH_AUTH_SOCK=/run/user/1000/keyring/ssh.XDG_SESSION_ID=2.XDG
GREETER_DATA_DIR=/var/lib/lightdm/data/user.USER=user.DESKTOP_SESSION=mate.PWD=/home/user/Aulas/Aula10.HOME=/home/user.SSH_A
GENT_PID=1207.QT_ACCESSIBILITY=1.XDG_SESSION_TYPE=x11.XDG_DATA_DIRS=/usr/share/mate:/usr/local/share:/usr/share/.MATE_DESKTO
P_SESSION_ID=2.this-is-deprecated.XDG_SESSION_DESKTOP=mate.GTK_MODULES=gail:atk-bridge:canberra-gtk-module:CLUTTER_BACKEND=x11.
TERM=xterm.SHELL=/bin/bash.VTE_VERSION=4601.XDG_SEAT_PATH=/org/freedesktop/DisplayManager/Seat0.XDG_CURRENT_DESKTOP=MATE.GPG
AGENT_INFO=/run/user/1000/gnupg/S.gpg-agent:0:1.QT_LINUX_ACCESSIBILITY_ALWAYS_ON=1.SHLVL=1.XDG_SEAT=seat0.LANGUAGE=en_US:en.W
INDOWID=8388614.GDMSESSION=mate.LOGNAME=user.DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus.XDG_RUNTIME_DIR=/run/user/
1000.XAUTHORITY=/home/user/.Xauthority.XDG_SESSION_PATH=/org/freedesktop/DisplayManager/Session0.PATH=/var/local/pycharm-comm
unity/bin:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games.SESSION_MANAGER=local/CSI:@/tmp/.ICE-unix/1178,unix/CSI:/t
mp/.ICE-unix/1178.GCC_COLORS=error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01.OLDPWD=/home/user/Aulas._./R
eadOverflow..ReadOverflow.....00.....y 40.....0000000..0.....(0..000Z.....000..000Z.....04/.....
.A~

```

Para produzir este output, declarou-se que se ia introduzir uma frase com o tamanho máximo que um inteiro pode tomar (2147483647) afim de obter mais informação.

5

Sabendo que 0x61626364 corresponde a "abcd"em ASCII, e como dados são guardados em memória no formato little-endian, é possível obter "dcba". Após obter o que se pretende, é necessário colocar esse valor na variável *control*, e pode-se fazer isso colocando um valor qualquer antes do texto, neste caso foi utilizado um conjunto de a's.

```

user@CSI:~/Aulas/Aula10$
user@CSI:~/Aulas/Aula10$ ./1-match aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaadcba
You win this game if you can change variable control to the value 0x61626364'
Congratulations, you win!!! You correctly got the variable to the right value
user@CSI:~/Aulas/Aula10$

```

6

Depois de ter sido compilado o programa com a flag -g. Foi chamado o gdb com o executável gerado para descobrir o endereço da função win() através do comando "p win". O endereço retornado foi "0x555555554740", passando os valores em hexadecimal para caracteres através da tabela ACSII resulta a frase "UUUUG@". Na stack do programa ficam registados os seguintes endereços de cima para baixo: declaração fp, declaração buffer, return, base pointer, atribuição fp e atribuição do buffer. Para que seja chamada a função win será necessário escrever no return o endereço da função win. como os índices do buffer crescem de baixo para cima, o seu ultimo índice fica abaixo da atribuição da variável fp. Assim ao preencher o buffer se utilizarmos 72 caracteres seguidos de "@UUUUU"uma vez que a arquitetura é little endian, serão preenchidos 64 indices do buffer 4 bytes da declaração do fp, 4 bytes do base pointer, e finalmente será preenchido o endereço de return com o endereço da função win().

7

Este exercício assemelha-se ao exercício anterior no sentido em que é necessário exceder os limites de um buffer de tamanho 64 afim de aceder ao endereço de retorno da função win. O primeiro passo será o de descobrir o endereço da função win que é 0x5555555546f0. Traduzindo este registo para código ASCII resulta na frase "UUUUFð"e conclua-se este exploit ao escreverem-se 72 carateres e acrescentando-se ao sufixo a frase na ordem inversa ("ðFUUUU")

uma vez que a arquitetura é little endian. Contudo, não se conseguiu alterar o endereço de retorno para o pretendido dado pois `ð` é um carácter ASCII especial e ocupa 2 bytes ao invés do convencional, tornando impossível alterar o endereço de retorno para `0x5555555546f0`.