CHEATSHEET SISHARD

COMANDO	REGISTRADORES	TRADUÇÃO	ANOTAÇÕES
add	Nesi, Ned	a+ub	Realiza a adição dos dois registadores e armazena em Yudi
call	duncace		Chama una funcac. OBS: antes de cada call devemos preencher os argumentos nos registradores convetos.
inut	Nesi, Nesi		Resiza multiplicacao signed
ion	(Ned, Neds, 2), Ness	66 a 442°C	Neste caso usado para arbmetica simples
nov	\$0x1, %mi	aut	guards \$0x1 em a
nov	Nesi, Nesi	4+6	guarda o valor de a em b
nov	Nesi, (Nesi)	194.0	pomino b aponta pra a
mov	Oxdea1(Notp),Neex		Paga o valor da variavel giobal guardado no endereco (besa1+1/s/p) e armazena en Natas
nst		return ret	Retorna o valor amuzierado nos reg de retorno. Equivale ao return
sub	Nesi, Nesi	a-a	Resiza a subtracao dos dois registradores e guarda no segundo
mov	(1649, 1646	4-5	
cmp	Nesi, Ned		compara valores e/ou registradores através da subtracao a-b. É acompanhado de albuma instruzao de condicao. Direita orimeiro
a et	166		val consultar e amuzienar o valor da flag no registrador al no byte menos significacio, n aberando os outros. Usado sembre dos de cino ou test
movabil	Nat, Nasa	net = (int) ret	vai realizar um casting de maneira adequada, evitando armapenar fixo de memoria no realizzador
atr	Yes	ed is 2 ed a ed sort	Manda sodos os bits pra direita, ou seja, mesmo que dividir por 2
1001	Nesi, Nesi		Compara a e b através do alib. Nibis utilizado para testar se o valor da variável é zero ou nilio.
1067	105,105		testa se a variavel fúsi ta zerada ou não
sect je	\$0x1, rei	# (b&&1++0)(
ion	Delifforni, Nordi, 2) , Norax	nt = 8+b+a*3	A operação LEA nunca acessa a memória, apenas move o resultado deste cálculo para o registrador destino.
PILHAS			
push	Nex		coloca flax no topo da pilha
pop	Nex		coloca o que estiver no topo da pilha, em dox
nab	\$0x10, Nasp		cria um espaco para as variaveis locais de 16bytes. O rep passa a apontar para o andereco rep-16
ion	Over/Norsept, Norsel		val guardar o endereco de rope 14 para b
add	\$0x10, 5ksp		remove as as variaveis criadas por sub-da pilha
CALCULO DE EN	ренесов		
mod	04,04(6rp)	in sart = 2	move o valor 2 para o endereco rap+6 da pilha
isa	Oul(Nonp), Noti	aubort	calcula o endereco da variavel, e prepara o registrador para a chamada de uma faccas
ARRAYS			5.5
mod	Statis, (Nation, Nation, 4)	vec[] = 5	Atibul o valor 5 para o elemento i (guardado em rox) do vetor veo (guardado em rox
nov	(Neds, Nesi, 4), Ness	ne - vec[]	move o valor de veciji para o registrador de setomo set
ion	(Node, Notel, 4), Neas	nt - Avecil	move o endereco de veci∏ pra ret

Tps	1/m	Donlo	Hexadecimal	Dini
		-108 o 127		000
	4	-\$10"ABHRUSTO'ABHO		
	2	4364237	1	000
	4	4 NOVABRIADADADAS		001
el der		1425	- 4	001
d	4	114200720	3	001
eling	4	144269726		010
cour	12	11636	4	010
	4	381107630107	5	010
	4	17x10*x34x10**		

BASE 16 <-> BASE 2

BASE 10 -> BASE 2

			0.444	٦
			Unive	•
1001	Investo ca bita	110 - 01	Paradelesis	.7
100	inverte ca bits	Some 1	Para decimal	†
Others bit is 1				
_				_

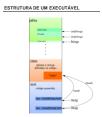
64bits	325/ts	16084	Strike	
long ou pointer	int ou unsigned	short ou unsigned	char ou	ARGUMENTO
16dS	Yes			
Yesi	Neci			ь
Nete	Neds	1604	168/168	4
Nex	Nece	No	Not Not	d
54	166			
500	1669			1
Nex	Ness	Nac	Nah ou Nat	100
1Mp				
Nesp	Nesp	Nep	Napi	

SET		
Inetrução	Traducao	Descrição
5450		Equal/Zero
sette	1=0	Not Equal Not Zero
0.005	40	(signed Negativo
sens	10	(signed) Não Negativ
set	*	(signed) Less than
sete	44	(signed) Less than o equal
setpe	in in	(signed) Greater than or equal
640		(signed) Greater than
seb	4	(unsigned) Below
nata.		Continued Show

JUMP		
Instrução	Traducao	Descrição
jmp		incondicional
- 1	0	Equal Sec
jne	140	not Equal not Zero
36	40	(signed Negativo
jn	>0	(signed) Não Negativo
		(signed) Less than
ja .	44	(signed) Less than or equal
ise	24	(signed) Greater than or equal
is	-	(signed) Greater than
Þ	-	(unsigned) Bellow
ja .		(unsigned) Above

CHEATSHEET - GDB DEBUGGING

gdb	inicia o debugging	
Sie arquivo	inicia o ampiivo no geb	
into	Retorna todas as possiveis combinacoes de info	
into registers. NLD	Itá retornar o valor armazenado no registrador ND	
info functions	retorna a lista de funcoes do programa	
break function	adciona um brealgoist na funcao	
run .	roda o programa	
srepi	Ita executar uma instrucao de maquina	
disas function	ira mostrar a lista de intrucces de function. Se tiver rodando o programa uma seta irá indicar a proxima instrucao a ser executada	
x advess.	Mostra o que esta armazienado no adress (endereco em hexa)	
sis adress	Mostra o que está no armazenamento se for char	
oli adress	Mostra o que está no armazenamento se for int	
quit	sal do gdb	
into variables	retorna lista de variaveis globals declaradas no programa	
print variavel	retorna o valor da variavel	
continue	continuară rodando o programa até que o prûximo breakpoint seja alcançado ou até que o programa termine.	



EXEMPLOS

SCANF COMANDO	REGISTRADORES	TRADUÇÃO	ANOTAÇÕES
ina	Osc/Nesp), Nesi	integ	Guarda o endereco da variavel local no funsi (segundo parametro do scarti)
ina	Ownership that includes		Carrega o formatador em Yurdi (primeiro parametro do scarti)
mov	\$040,New		só para dar um "reser", para zerar o registrador de retorno e debar o scard retornar also caso sela necessário
caliq	0x1040 <_lacciti_scart@ptx	scant("Nd", Am)	OSG: Para saber o que tem no formatador consultar atravé do comeando als adress no gdb.

SE	REGISTRADORES	TRADUÇÃO	ANOTAÇÕES	ARRAYS	REGISTRADORES	TRADUÇÃO	ANOTAÇÕES
	\$0x14,%eax		-	ina	Ovo(Nirsp),Nirsi	in no	Guarda o endereco da variavel local no Yursi (segundo parametro do scarifi
	Out 1a0 vlanot+50+	I (100 > 20)	Condição para executar as instruções do if. Se não estiver nessa condição, salta para a linha do elos.	ina	Datab(Nirg), Nirdi # Da2006		Carrega o formatador em Yurdi (primeiro parametro do sca
				mov	\$0x0,%eas		
				caliq	Ox1040 <_isoc99_scan58.pt>	scard("Nd", Am)	OSG: Para saber o que sem no formatador consultar atrave do comeando als adress no gilb.

WHILE/F	FOR REGISTRADORES	TRADUÇÃO	ANOTAÇÕES
тр	Nesi Nesx		
-		white (ratio ess)	Mesma instrução do il, entretarso, ocorre um pulo pra comparação novamente
dd	\$0x1,56esc		Increments do loop
тр	Out! «funcivil»		Salto pra comparação novamente

TIPOS SCANF E PRINTF

COMMISS	TPO
%6	char
Na	axing
101	signed short
Nhu	unsigned short
%d ou %i	int
No	unsigned int
10 ou 16e	foat
No	memory address
16 or 167 ou 165	long
162	long double
194	unsigned int or unsigned double
100	double