

Progettazione ottimizzazione DETON

Definito un registro r la metrica H_r definisce inversamente da quanto tempo un dato non è stato scritto in r , quando un dato è scritto in r H_r è impostato a un valore H_{max} anche detto heat, più è alto H_r meglio è.

Definito un programma p e la sua versione ofuscata p_0 la metrica E_p-p_0 definisce la differenza in istruzioni Assembly tra p e p_0 , più è alto E_p-p_0 più è alto l'overhead introdotto da Deton.

Parametri DETON

path/to/json

cfg entry point

a initial heat value

b garbage insertion value

c garbage block size

d obfuscate value

e register scrambling value

path/to/output

path/to/metrics_output

b,c,d introducono overhead

a,e non introduce overhead

Cromosoma

Codifica delle soluzioni con rappresentazione a numeri interi

b =50	c =4	d =50	e =50	a =50
-------	------	-------	-------	-------

Creazione popolazione iniziale

La popolazione iniziale viene creata coprendo in maniera randomica ma uniformemente il range dei parametri b (0, 200), d (0, 200), e (0, 300). Il parametro c viene generato anch'esso in maniera randomica ma nel range (3, 8).

[Il parametro \$a\$ invece è fisso a 32 ma va sviluppata una procedura per il calcolo del valore ideale.](#)

Funzione Fitness

Punteggio: Associa ad ogni individuo un punteggio, il punteggio è definito dalla somma di due funzioni ($punt_over$ e $punt_heat$), $punt_over$ parte dal valore di overhead e assegna un punteggio compreso tra -10000 e +10000 mediante il seguente criterio: definito un valore O_max variabile per ogni programma in base al numero iniziale di istruzioni prima della manipolazione di DETON si associa il valore massimo di punteggio se l'overhead è 1%, il punteggio assegnato scende man mano che ci si avvicina a O_max , una volta superata la soglia di O_max il punteggio diventa negativo crescendo fino a un massimo di -10000 punti con un overhead di 2 O_max , qualsiasi overhead superiore restituisce sempre -10000 punti. La seconda funzione $punt_heat$ associa un punteggio compreso tra 10000 e

-10000 mediante il seguente criterio: definita la media del valore di heat per ogni registro si divide il punteggio massimo tra tutti i registri e più la media è vicina al heat ideale più si assegna un punteggio alto, al valore di $\text{heat_ideale}/2$ si assegna punteggio 0 e scendendo ancora il punteggio diventa negativo. Si sommano tutti i punteggi dei vari registri e si ottiene il punteggio totale.

Funzione selezione parentale:

$N=1000$ $N+1 < -100$ nella successiva
400 migliori e li faccio accoppiare per ottenerle 800 $<- N+1$
100 presi tra i peggiori $<- N+1$
Ora che ne ho 1000 applico la mutazione

Mutazione

Bit flip mutation di geni selezionati in maniera casuale, la probabilità di mutazione cresce con il diminuire della posizione in classifica dell'individuo
Almeno un 5% degli individui subiscono la mutazione in ogni generazione

Crossover

Crossover a più punti, ogni madre e padre generano 4 figli, il crossover avviene rispettivamente tra i parametri: (heat, garbage, garbage_block) (scrambling) (obfuscate)