

TD/TP6 : Tableaux & Structures de données

Nga Nguyen, EISTI

Exo 1 : Structure et Alignement

Voici une structure en C:

```
struct {
    char    *a;
    short   b;
    double  c;
    char    d;
    float   e;
    char    f;
    long    g;
    int     h;
} rec;
```

1. Quels sont les offsets en octet des membres de cette structure ?
2. Quelle est la taille totale de la structure ?
3. Réarranger les membres pour minimiser l'espace mémoire gaspillé.

Exo 2 : Structure et Union

Étudier l'alignement et la taille des définitions suivantes :

```
typedef struct {
    char c;
    double *p;
    int i;
    double d;
    short s;
} struct1;
```

```
typedef union {
    char c;
    double *p;
    int i;
    double d;
    short s;
} union1;
```

Exo 3 : Tableau trois dimensions

Considérer le code suivant, où R, S et T sont des constants déclarés avec #define.

```
long A[R][S][T];
long store-ele(long i, long j, long k, long *dest) {
    *dest = A[i][j][k];
    return sizeof(A);
}
```

Et son code assembler:

```
// i en %rdi, j en %rsi, % k en %rdx, dest en %rcx
store_ele:
    leaq    (%rsi, %rsi, 2), %rax
    leaq    (%rsi, %rax, 4), %rax
    movq    %rdi, %rsi
    salq    $6, %rsi
    addq    %rsi, %rdi
    addq    %rax, %rdi
    addq    %rdi, %rdx
    movq    A(,%rdx,8), %rax
    movq    %rax, (%rcx)
    movl    $3640, %eax
    ret
```

Utiliser votre talent de reverse engineering pour trouver la valeur de R, S et T.