

Heap Message

- * Traditional exploitation has become tougher
- * On Linux... who knows how to do it?
- * On Windows
 - * XP SP2 made it tougher
 - * Lookaside overwrites
 - * Last chunk overwrite
 - * Vista anybody?

Heap Exploitation

* Vista:

* Lookaside lists are gone...

* Welcome Low Fragmentation Heap (?)

* ... and more security checks.



```
and [esi+_HEAP.Encoding], 0
or byte ptr [esi+52h], 10h
mov eax, [esi+_HEAP.Encoding]
mov [esi+4Ch], eax
call _RtlpHeapGenerateRandomValue64@0 ; RtlpHeapGenerateRandomValue64()
or [esi+_HEAP.Encoding], eax
call _RtlpHeapGenerateRandomValue64@0 ; RtlpHeapGenerateRandomValue64()
mov word ptr [esi+(_HEAP.Encoding+4)], ax
mov byte ptr [esi+56h], 0
mov byte ptr [esi+57h], 0
call _RtlpHeapGenerateRandomValue64@0 ; RtlpHeapGenerateRandomValue64()
mov [esi+58h], eax
```

* ...



```
mov eax, [ebx+_HEAP.Encoding]
xor dword ptr [esi+_HEAP_ENTRY.Size], eax
mov al, [esi+_HEAP_ENTRY.Flags]
xor al, byte ptr [esi+(_HEAP_ENTRY.Size+1)]
xor al, byte ptr [esi+_HEAP_ENTRY.Size]
cmp [esi+_HEAP_ENTRY.SmallTagIndex], al
jnz loc_776CDCB1
```

Heap Exploitation

- So... are heap exploits gone?
 - No way! we love them! help them stay!
- * So lets go back to basics...

*What's after the vulnerable buffer,
which if corrupted, will let an attacker
gain code execution?*

anybody?

Heap Massaging

- * Code
- * Sensitive information
- * Pointers, structures, etc. that let us do a 4bw
- * Class pointers

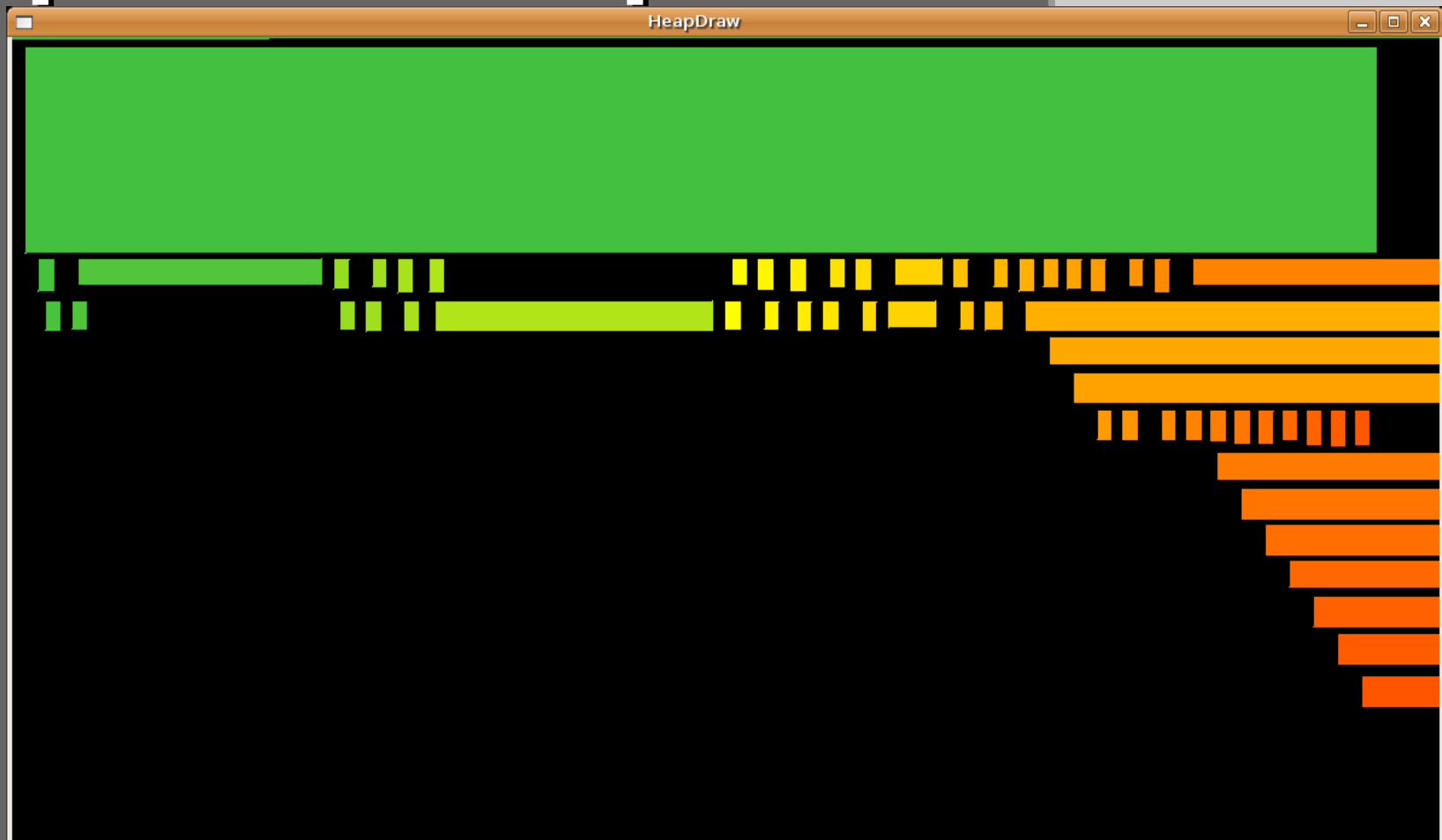
Protections

C++ Usage

Heap Massaging

- * Heap block ordering is vital.
- * We can't leave it to fate.
- * Need to take control of the order of blocks.
- * Learn more of the application.
- * Find other commands that you can use.
- * Find memory leaks.
- * Ideally, build remote malloc(), free(), etc.

HeapDraw/HeapTracer



Limits: 1.082366-1.125270 0x0004ee1f-0x0004f327 (1288)

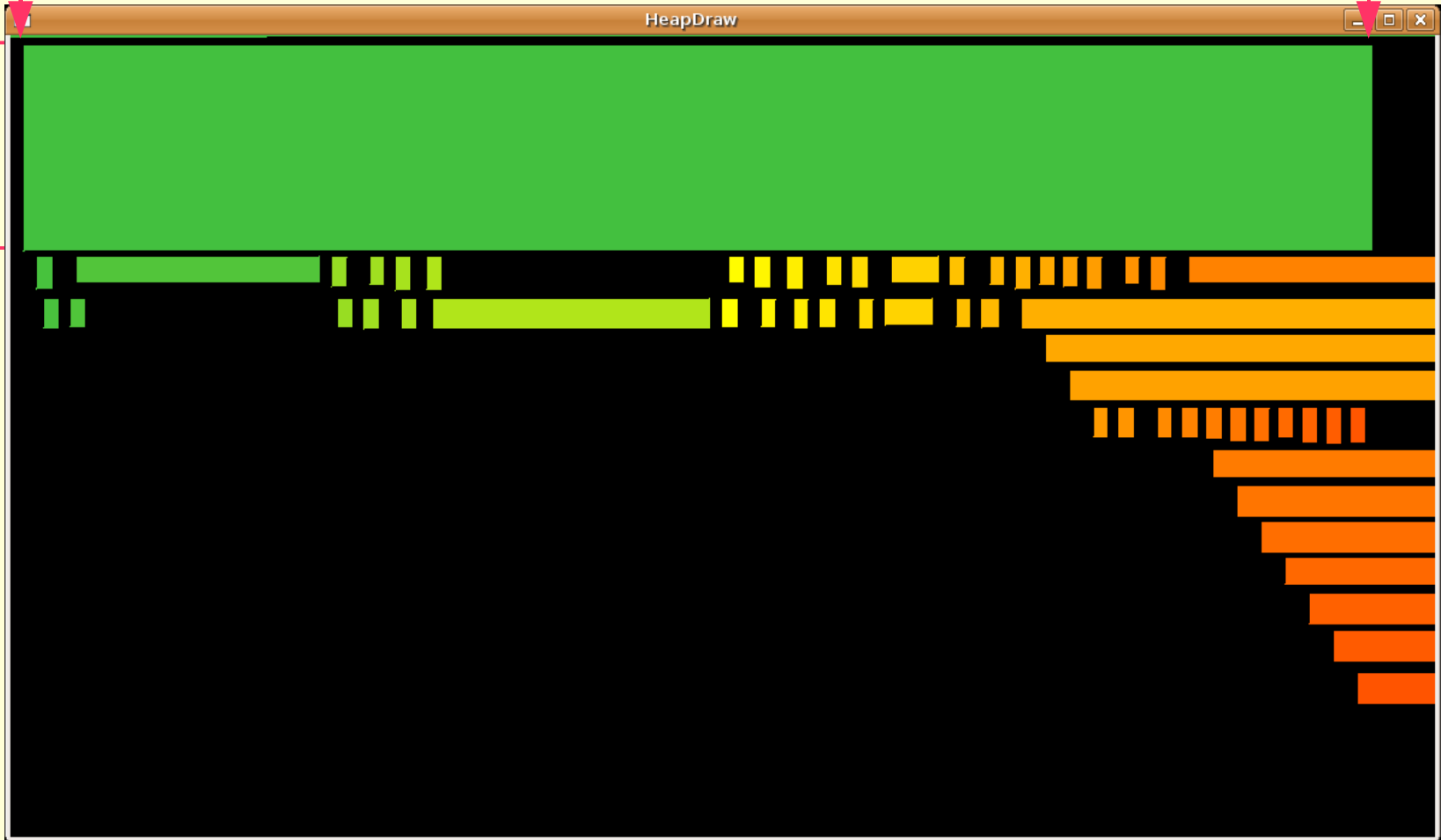
malloc()

free()

Time

size

Address



Limits: 1.082366-1.125270 0x0004ee1f-0x0004f327 (1288)

Heap Massaging

- * google for “Heap Feng Shui” (Alex Sotirov)
- * HeapDraw/HeapTracer and a few more tools
(soon) <http://oss.coresecurity.com>