

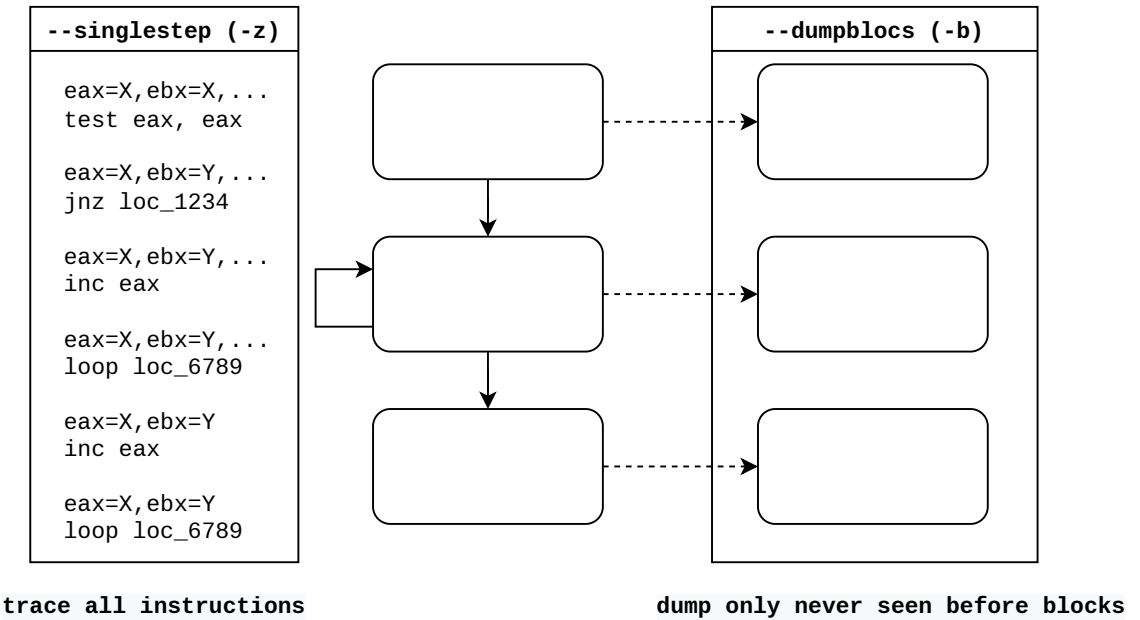
```
from miasm.analysis.sandbox import Sandbox_Win_x86_32
from miasm.core.locationdb import LocationDB

# Parse arguments
parser = Sandbox_Win_x86_32.parser(description="PE sandboxer")
parser.add_argument("filename", help="PE Filename")
options = parser.parse_args()

# Create sandbox
loc_db = LocationDB()
sb = Sandbox_Win_x86_32(loc_db, options.filename, options, globals())

# Run
sb.run()
```

Logging



Emulation cursor: segmentation

Default, useful for shellcodes



```
mov eax, 0x30
mov ebx, FS:[eax]

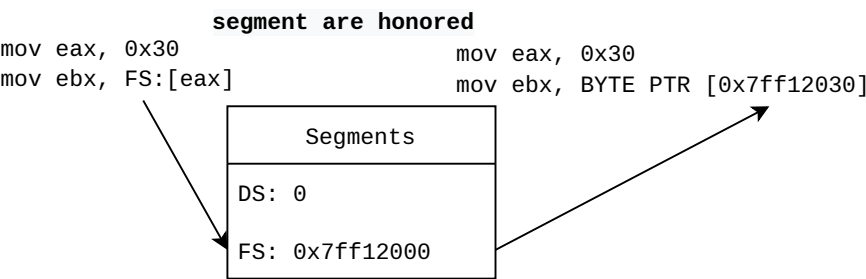
mov eax, 0x30
mov ebx, BYTE PTR [0x30]
```

segment are NOT honored

With segments, useful for 16bits or import-by-hash

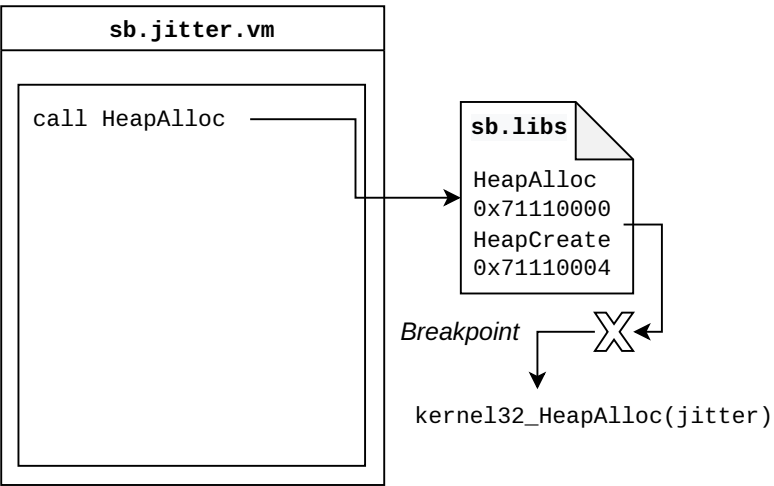


```
-s, --usesegm
```



Emulation cursor: dependencies

Sandbox default, breakpoint set for imports, to emulate them

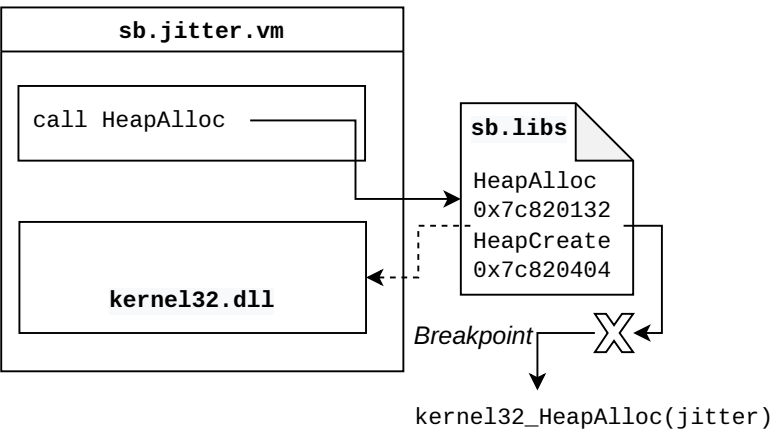


DLL are loaded in virtual memory.



```
-i, --dependencies (real dependencies)
```

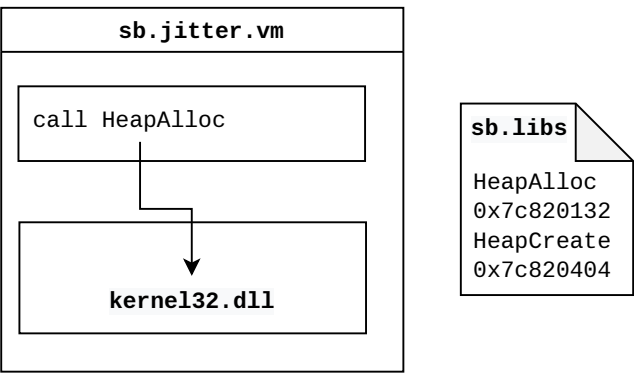
```
-l, --loadbasedll (hardcoded list)
```



Remove breakpoint to use the in-memory function



```
sb.jitter.remove_breakpoints_by_address(
    sb.libs.cname2addr["ntdll_swprintf"]
)
```

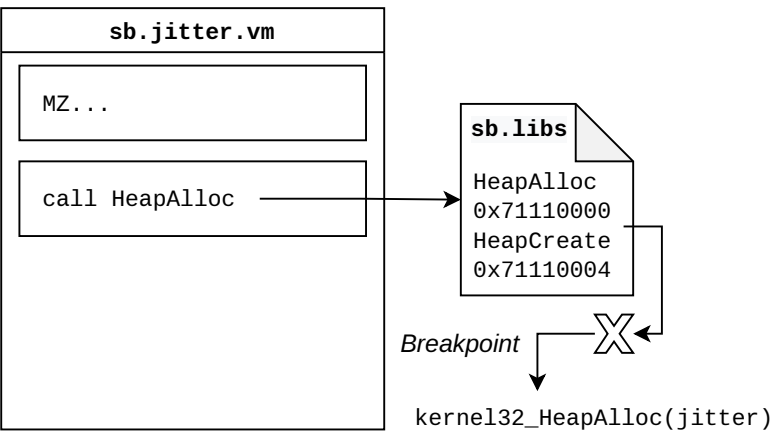


Emulation cursor: structures

PE header is loaded in virtual memory.



```
-o, --load-hdr
```



Some Windows structures are created and filled



```
-y, --use-windows-structs
```

