



## ***sensor***

```
-observers: set<sensor_observ  
-sensing: bool = false  
-status: unsigned short = 0
```

```
+<<virtual>> sensing_status()  
+<<virtual>> sensing(): bool  
+<<virtual>> start_sensing():  
+<<virtual>> stop_sensing():  
+<<virtual>> toggle_sensing()  
+<<virtual>> sense(): void  
+attach_observer(observer:sen  
+detach_observer(observer:sen
```

.
er>
: int  int int : int  sor_observer &): void sor_observer &): void



```
+detach_observer(observer:Sensor)
+observer_count(): int
```



-----

## **Syscall\_Sensor** *singleton*

```
-FTRACE_DIR: string = /sys/kernel/trace
-TRACING_ON_FILE: string = trace_pipe
-SYS_ENTRY_DIR: string = syscalls
-SYS_EXIT_DIR: string = syscalls
-EVENTS_DIR: string = events
-TRACE_PIPE_FILE: string = trace_pipe
-CURRENT_TRACER_FILE: string = current_tracer
-trace_pipe_stream: fstream
-ENABLE_FILE: string = enable
```

```
-<<constructor>> Syscall_Sensor
+get_instance(): Syscall_Sensor
+sensing_status(): int
+sensing(): bool
+start_sensing(): int
+stop_sensing(): int
+toggle_sensing(): int
-set_tracer(tracer:string="no_tracer")
-set_tracing_on(on:bool): int
```

```
    sub_observer &); void
```

```
nsor
```

```
    rnel/debug/tracing/  
    acing_on  
    enter/  
    xit/
```

```
    ace_pipe  
    = current_tracer
```

```
or()  
or &
```

```
p"): void  
igned short
```