

```

1  --- yama_lsm_old.c      2027-12-02 05:16:29.036771603 -0500
2  +++ yama_lsm.c2027-12-02 05:16:11.124770882 -0500
3  @@ -430,11 +430,39 @@
4
5          return rc;
6  }
7  +//custom_hook for preventing the setimte in the past by the user. This takes care of the null
inputs and any input that is in the past of the current date and time.
8
9  +int yama_settime(const struct timespec64 *ts, const struct timezone *tz)
10 +{
11 +
12 +
13 +    struct timespec64 now;
14 +
15 +    //if the ts is set to null or the user has tried to pass a null value to set time; Effort to
stop the kernel to go into panic mode.
16 +    if (!ts) {
17 +        pr_info("Attempt to set null time.\n");
18 +        return -EFAULT;
19 +    }
20 +
21 +
22 +    getnstimeofday64(&now);
23 +
24 +    // Condition to check if the new time is before the current time.
25 +    // We return -EPERM to deny.
26 +    if (timespec64_compare(ts, &now) < 0) {
27 +        pr_info("rejecting time change in the past\n");
28 +        return -EPERM;
29 +
30 +    }

```

```
31 + // Everything is okay, return 0 to grant permission.
32 + pr_info("Granting permission for changing the time\n");
33 + return 0;
34 +}
35
36 static struct security_hook_list yama_hooks[] __lsm_ro_after_init = {
37     LSM_HOOK_INIT(ptrace_access_check, yama_ptrace_access_check),
38     LSM_HOOK_INIT(ptrace_traceme, yama_ptrace_traceme),
39 +    LSM_HOOK_INIT(settime, yama_settime),
40     LSM_HOOK_INIT(task_prctl, yama_task_prctl),
41     LSM_HOOK_INIT(task_free, yama_task_free),
42     };
```