Disassembly of login()

```
_check_secret:
0000000100000d40    subq    $0x38, %rsp
0000000100000d44    movq    %rdi, 0x28(%rsp)
0000000100000d49    movl    %esi, 0x24(%rsp)
0000000100000d4d    cmpl    $-0x1, 0x24(%rsp)
0000000100000d52    jne 0x100000d73
0000000100000d58    leaq    0x1ab(%rip), %rdi           ## literal pool for: "problem reading password.txt\n"
0000000100000d5f    movb    $0x0, %al
0000000100000d61    callq   0x100000ef2                 ## symbol stub for: _printf
0000000100000d66    movl    $0x0, 0x34(%rsp)
0000000100000d6e    jmp 0x100000dd5
0000000100000d73    leaq    0x1ae(%rip), %rax           ## literal pool for: "superSecretPassword"
0000000100000d7a    movq    %rax, 0x18(%rsp)
0000000100000d7f    movslq  0x24(%rsp), %rax
0000000100000d84    movq    %rax, 0x8(%rsp)
0000000100000d89    movq    0x18(%rsp), %rdi
0000000100000d8e    callq   0x100000f04                 ## symbol stub for: _strlen
0000000100000d93    movq    0x8(%rsp), %rcx
0000000100000d98    movq    %rax, %rdx
0000000100000d9b    xorl    %eax, %eax
0000000100000d9d    cmpq    %rdx, %rcx
0000000100000da0    movb    %al, 0x17(%rsp)
0000000100000da4    jne 0x100000dc8
0000000100000daa    movq    0x28(%rsp), %rdi
0000000100000daf    movq    0x18(%rsp), %rsi
0000000100000db4    movslq  0x24(%rsp), %rdx
0000000100000db9    callq   0x100000ee6                 ## symbol stub for: _memcmp
0000000100000dbe    cmpl    $0x0, %eax
0000000100000dc1    sete    %al
0000000100000dc4    movb    %al, 0x17(%rsp)
0000000100000dc8    movb    0x17(%rsp), %al
0000000100000dcc    andb    $0x1, %al
0000000100000dce    movzbl  %al, %eax
0000000100000dd1    movl    %eax, 0x34(%rsp)
0000000100000dd5    movl    0x34(%rsp), %eax
0000000100000dd9    addq    $0x38, %rsp
0000000100000ddd    retq
0000000100000dde    nop
_success:
0000000100000de0    subq    $0x18, %rsp
0000000100000de4    movq    _sh(%rip), %rax
0000000100000deb    movq    %rax, (%rsp)
0000000100000def    movq    $0x0, 0x8(%rsp)
0000000100000df8    leaq    0x13d(%rip), %rdi           ## literal pool for: "successful login!\n"
0000000100000dff    callq   0x100000ef8                 ## symbol stub for: _puts
0000000100000e04    movq    _sh(%rip), %rdi
0000000100000e0b    movq    %rsp, %rsi
0000000100000e0e    movq    0x1f3(%rip), %rax           ## literal pool symbol address: _environ
0000000100000e15    movq    (%rax), %rdx
0000000100000e18    callq   0x100000ee0                 ## symbol stub for: _execve
0000000100000e1d    addq    $0x18, %rsp
0000000100000e21    retq
0000000100000e22    nopw    %cs:(%rax,%rax)
_failure:
0000000100000e30    pushq   %rax
0000000100000e31    leaq    0x117(%rip), %rdi           ## literal pool for: "wrong password\n"
0000000100000e38    callq   0x100000ef8                 ## symbol stub for: _puts
0000000100000e3d    popq    %rax
0000000100000e3e    retq
0000000100000e3f    nop
```

00111000

```
_login:
0000000100000e40    subq    $0x38, %rsp
0000000100000e44    leaq    0x114(%rip), %rdi           ## literal pool for: "password.txt"
0000000100000e4b    xorl    %esi, %esi
0000000100000e4d    movb    $0x0, %al
0000000100000e4f    callq   0x100000eec                 ## symbol stub for: _open
0000000100000e54    movl    %eax, 0xc(%rsp)
0000000100000e58    leaq    0x10d(%rip), %rdi           ## literal pool for: "enter your password:\n"
0000000100000e5f    movb    $0x0, %al
0000000100000e61    callq   0x100000ef2                 ## symbol stub for: _printf
0000000100000e66    movl    0xc(%rsp), %edi
0000000100000e6a    leaq    0x10(%rsp), %rsi
0000000100000e6f    movl    $0x3e8, %edx                ## imm = 0x3E8
0000000100000e74    callq   0x100000efe                 ## symbol stub for: _read
0000000100000e79    movl    %eax, 0x8(%rsp)
0000000100000e7d    movl    0xc(%rsp), %edi
0000000100000e81    callq   0x100000eda                 ## symbol stub for: _close
0000000100000e86    leaq    0x10(%rsp), %rdi
0000000100000e8b    movl    0x8(%rsp), %esi
0000000100000e8f    callq   _check_secret
0000000100000e94    addq    $0x38, %rsp
0000000100000e98    retq
0000000100000e99    nopl    (%rax)
_main:
0000000100000ea0    pushq   %rax
0000000100000ea1    movl    $0x0, 0x4(%rsp)
0000000100000ea9    callq   _login
0000000100000eae    movl    %eax, (%rsp)
0000000100000eb1    cmpl    $0x0, (%rsp)
0000000100000eb5    je  0x100000ec5
0000000100000eba    callq   _success
0000000100000ec0    jmp 0x100000eca
0000000100000ec5    callq   _failure
0000000100000eca    leaq    0xb1(%rip), %rdi            ## literal pool for: "exiting in main\n"
0000000100000ed1    callq   0x100000ef8                 ## symbol stub for: _puts
0000000100000ed6    xorl    %eax, %eax
0000000100000ed8    popq    %rcx
0000000100000ed9    retq
```

— function Prologue. Subtracts 0x38 (56 in decimal) from the stack pointer