

```
; ===== BEGINNING OF PROCEDURE =====  
  
; Variables:  
; var_8: -8  
; var_C: -12  
; var_10: -16  
; var_14: -20  
; var_18: -24  
; var_20: -32  
; var_28: -40  
; var_2C: -44  
; var_30: -48  
; var_31: -49  
; var_32: -50  
; var_38: -56  
; var_3C: -60  
; var_40: -64  
; var_44: -68  
; var_48: -72  
; var_4C: -76  
; var_50: -80  
; var_54: -84  
; var_58: -88  
; var_5C: -92  
  
_main:  
push    rbp  
mov     rbp, rsp  
sub     rsp, 0x60  
lea     rax, qword [_array1] ; _array1  
mov     rcx, qword [__stack_chk_guard_100001010] ; __stack_chk_guard_100001010  
mov     rcx, qword [rcx]  
mov     qword [rbp+var_8], rcx  
mov     dword [rbp+var_14], 0x0  
mov     dword [rbp+var_18], edi  
mov     qword [rbp+var_20], rsi  
mov     rcx, qword [_secret] ; _secret  
sub     rcx, rax  
mov     qword [rbp+var_28], rcx  
mov     dword [rbp+var_30], 0x17  
mov     dword [rbp+var_2C], 0x0
```

```
loc_100000d07:  
movsxd  rax, dword [rbp+var_2C] ; CODE XREF=_main+111  
cmp     rax, 0x20000  
jae     loc_100000d34
```

```
lea     rax, qword [_array2] ; _array2  
movsxd  rcx, dword [rbp+var_2C]  
mov     byte [rax+rcx], 0x1  
add     eax, 0x1  
mov     dword [rbp+var_2C], eax  
jmp     loc_100000d07
```

```
loc_100000d34:  
cmp     dword [rbp+var_18], 0x3 ; CODE XREF=_main+81  
jne     loc_100000d8d
```

```
lea     rsi, qword [aP] ; argument "format" for method imp__stubs__scanf, "%p"  
lea     rax, qword [rbp+var_28]  
mov     rcx, qword [rbp+var_20]  
mov     rdi, qword [rcx+8] ; argument "s" for method imp__stubs__scanf  
mov     rdx, rax  
mov     al, 0x0  
call    imp__stubs__scanf ; scanf  
lea     rsi, qword [aD] ; argument "format" for method imp__stubs__scanf, "%d"  
lea     rdx, qword [rbp+var_30]  
lea     rcx, qword [_array1] ; _array1  
mov     rdi, qword [rbp+var_28]  
sub     rdi, rcx  
mov     qword [rbp+var_28], rdi  
mov     rcx, qword [rbp+var_20]  
mov     rdi, qword [rcx+0x10] ; argument "s" for method imp__stubs__scanf  
mov     dword [rbp+var_38], eax  
mov     al, 0x0  
call    imp__stubs__scanf ; scanf  
mov     dword [rbp+var_3C], eax
```

```
loc_100000d8d:  
lea     rdi, qword [aReadingDBytesn] ; argument "format" for method imp__stubs__printf, "Reading %d bytes:\\n", CODE XREF=_main+120  
mov     esi, dword [rbp+var_30]  
mov     al, 0x0  
call    imp__stubs__printf ; printf  
mov     dword [rbp+var_40], eax
```

```
loc_100000da1:  
mov     eax, dword [rbp+var_30] ; CODE XREF=_main+483  
add     eax, 0xffffffff  
mov     dword [rbp+var_30], eax  
cmp     eax, 0x0  
jl      loc_100000ea8
```

```
lea     rdi, qword [aAttamptingConf] ; argument "format" for method imp__stubs__printf, "Attampting Confirmation %p "  
mov     rsi, qword [rbp+var_28]  
mov     al, 0x0  
call    imp__stubs__printf ; printf  
rdx, qword [rbp+var_10] ; argument #3 for method _readMemoryByte  
lea     rsi, qword [rbp+var_32] ; argument #2 for method _readMemoryByte  
mov     rdi, qword [rbp+var_28] ; argument #1 for method _readMemoryByte  
mov     rcx, rdi  
add     rcx, 0x1  
mov     qword [rbp+var_28], rcx  
mov     dword [rbp+var_44], eax  
call    _readMemoryByte ; _readMemoryByte  
lea     rdi, qword [aS] ; argument "format" for method imp__stubs__printf, "%s: "  
lea     rdx, qword [aUnconfirmed] ; "Unconfirmed"  
lea     rdx, qword [aConfirmed] ; "Confirmed"  
mov     eax, dword [rbp+var_10]  
mov     r8d, dword [rbp+var_C]  
shl     r8d, 0x1  
cmp     eax, r8d  
cmovge rcx, rdx  
mov     rsi, rcx  
mov     al, 0x0  
call    imp__stubs__printf ; printf  
movzx   esi, byte [rbp+var_32]  
movzx   r8d, byte [rbp+var_32]  
cmp     r8d, 0x1f  
mov     dword [rbp+var_48], eax  
mov     dword [rbp+var_4C], esi  
jle     loc_100000e47
```

```
movzx   eax, byte [rbp+var_32]  
cmp     eax, 0x7f  
jge     loc_100000e47
```

```
movzx   eax, byte [rbp+var_32]  
mov     dword [rbp+var_50], eax  
jmp     loc_100000e54
```

```
loc_100000e47:  
mov     eax, 0x3f ; CODE XREF=_main+360, _main+373  
mov     dword [rbp+var_50], eax  
jmp     loc_100000e54
```

```
loc_100000e54:  
mov     eax, dword [rbp+var_50] ; CODE XREF=_main+386, _main+399  
movzx   esi, byte [rbp+var_32]  
lea     rdi, qword [a0x02xe2x80x99] ; argument "format" for method imp__stubs__printf, "0x%02X=\\x02\\x80\\x99c\\x02\\x80\\x99 score=%d "  
mov     ecx, dword [rbp+var_10]  
mov     esi, dword [rbp+var_4C]  
mov     edx, eax  
mov     al, 0x0  
call    imp__stubs__printf ; printf  
cmp     dword [rbp+var_C], 0x0  
mov     dword [rbp+var_54], eax  
jle     loc_100000e92
```

```
lea     rdi, qword [aAlt0x02xScored] ; argument "format" for method imp__stubs__printf, "(Alt: 0x%02X score=%d)"  
movzx   esi, byte [rbp+var_31]  
mov     edx, dword [rbp+var_C]  
mov     al, 0x0  
call    imp__stubs__printf ; printf  
mov     dword [rbp+var_58], eax
```

```
loc_100000e92:  
lea     rdi, qword [aN] ; argument "format" for method imp__stubs__printf, "\\n", CODE XREF=_main+436  
mov     al, 0x0  
call    imp__stubs__printf ; printf  
mov     dword [rbp+var_5C], eax  
jmp     loc_100000da1
```

```
xor     eax, eax  
add     rsp, 0x60  
ret
```

```
loc_100000ec7:  
call    imp__stubs__stack_chk_fail ; __stack_chk_fail, CODE XREF=_main+505
```

```
loc_100000ea8:  
mov     rax, qword [__stack_chk_guard_100001010] ; __stack_chk_guard_100001010, CODE XREF=_main+237  
mov     rax, qword [rax]  
mov     rcx, qword [rbp+var_8]  
cmp     rax, rcx  
jne     loc_100000ec7
```