

# C for Linux

## Session 3

BY:

Eng.Ahmed Teirelbar  
Software Engineer  
Shuja' Consulting



**EGLUG**

جنو لینکس مصر

# **gdb Basics**

- **-g**
- **In execution**
  - **run**
  - **continue**
  - **step**
  - **breakpoint**
  - **variables**
  - **directives**
  - **execute code**



**EGLUG**

جنو لینکس مصر

# Pointer Concepts

- An array is a pointer to the first element
- Pointer subscript access
- Scope issues – Back to the stack picture
- NULL pointer



**EGLUG**

جنو لینکس مصر

# Pointers

- SEGV
- Memory overwrites – “but it worked before!!!”
- null termination issues
- Core dump



**EGLUG**

جنو لینکس مصر

# Debugging and Detection

- More gdb
  - dumping memory
  - Casting
  - Watchpoints
- Detection
  - Sanity checks
  - Macros with sizeof
  - Magic numbers in Structures



**EGLUG**

جنتو لینکس مصر

# Allocation and Memory

- The Heap data structure
- Back to the process picture
- Very brief overview of system memory allocation
  - `brk()`
  - Granularity & page size
  - Allocation algorithms
  - Block data
  - Alignment
  - Swapping



**EGLUG**

جنتو لینکس مصر

# Memory Leaks

- What is it?
- Why do we care? what is the OOM killer?
- Detection
  - Your own code, macros
  - top and /proc/meminfo
  - Valgrind and profiling
  - <http://valgrind.org/>
  - <http://www.advancedlinuxprogramming.com/> (Appendix A.2)



**EGLUG**

جنو لینکس مصر

# Structures/Unions in Memory

- Alignment
  - Sizeof struct, union, pointer
- Casting
- Traversals of pointers



**EGLUG**

جنو لینکس مصر

# Functions & the Stack

- Overview
  - Base offset/Frame pointer
  - RET
  - Local variables
- objdump
- gdb & stack trace
  - where
  - up/down
  - Huge for core dumps
- Recursion & your stack



**EGLUG**

جنتو لینکس مصر

# The Infamous Stack Overflow Attack

- Overview of Concept
- How it works?
  - The vulnerability
  - Integrate the whole stack picture
  - How you place the exploiting code –high level
- How do you protect against it?
- Heap overflow



**EGLUG**

جنتو لینکس مصر

# Structures w/Functions:

## Data Abstraction

- Higher level of operation
  - Object and Interface
- Modularity of a defined interface
  - Easier implementation per piece
  - Unit test
  - Module-level behavioral Changes



**EGLUG**

جنو لینکس مصر

# Data Abstraction Cont'd

- Data Hiding
  - Extension of type
  - Type-level behavioral changes (decoupling)
- Can be taken too far – be sensible



**EGLUG**

جنو لینکس مصر