ngx_palloc.h Documentation:

Macros Defined:

```
#define NGX_MAX_ALLOC_FROM_POOL  (ngx_pagesize - 1)
#define NGX_DEFAULT_POOL_SIZE    (16 * 1024)
#define NGX_POOL_ALIGNMENT       16
#define NGX_MIN_POOL_SIZE
```

Functions Declared:

```
ngx_pool_t *  ngx_create_pool(  size_t       size,
                                ngx_log_t  *log
                              )

void ngx_destroy_pool(  ngx_pool_t *pool );

void  ngx_reset_pool(  ngx_pool_t *pool )

void *  ngx_palloc(ngx_pool_t *pool,
                   size_t        size)

void * ngx_pnalloc(  ngx_pool_t *pool,
                     size_t        size
                   )

void * ngx_pmemalign(  ngx_pool_t  *pool,
                       size_t        size,
                       size_t        alignment
                     )

ngx_int_t  ngx_pfree(  ngx_pool_t *pool,
                       void          *p
                     )

void * ngx_pcalloc(  ngx_pool_t   *pool,
                     size_t         size
                   )

ngx_pool_cleanup_t *  ngx_pool_cleanup_add(  ngx_pool_t  *p,
                                             size_t         size
                                           )
```

```
void  ngx_pool_run_cleanup_file(  ngx_pool_t  *p,
                                  ngx_fd_t    fd
                                )

void ngx_pool_cleanup_file(  void *data )

void ngx_pool_delete_file(  void *data)

void *ngx_alloc(  size_t     size,
                  ngx_log_t  *log
                )

void *ngx_calloc(  size_t      size,
                   ngx_log_t   *log
                 )

ngx_pool_t *ngx_create_pool(  size_t      size,
                              ngx_log_t   *log
                            )

void ngx_destroy_pool( ngx_pool_t *pool)

void ngx_reset_pool( ngx_pool_t *pool);



void *ngx_palloc(  ngx_pool_t  *pool,
                   size_t      size
                 )

void *ngx_pnalloc(  ngx_pool_t  *pool,
                    size_t      size
                  )

void *ngx_pcalloc( ngx_pool_t  *pool,
                   size_t      size
                 )

void *ngx_pmemalign( ngx_pool_t   *pool,
                     size_t       size,
                     size_t       alignment
                   )
```

```
ngx_int_t ngx_pfree( ngx_pool_t   *pool,
                     void         *p
                   )

ngx_pool_cleanup_t *ngx_pool_cleanup_add( ngx_pool_t  *p,
                                          size_t      size
                                        )

void ngx_pool_run_cleanup_file( ngx_pool_t  *p,
                                ngx_fd_t     fd
                              )

void ngx_pool_cleanup_file( void *data);

void ngx_pool_delete_file(void *data);
```

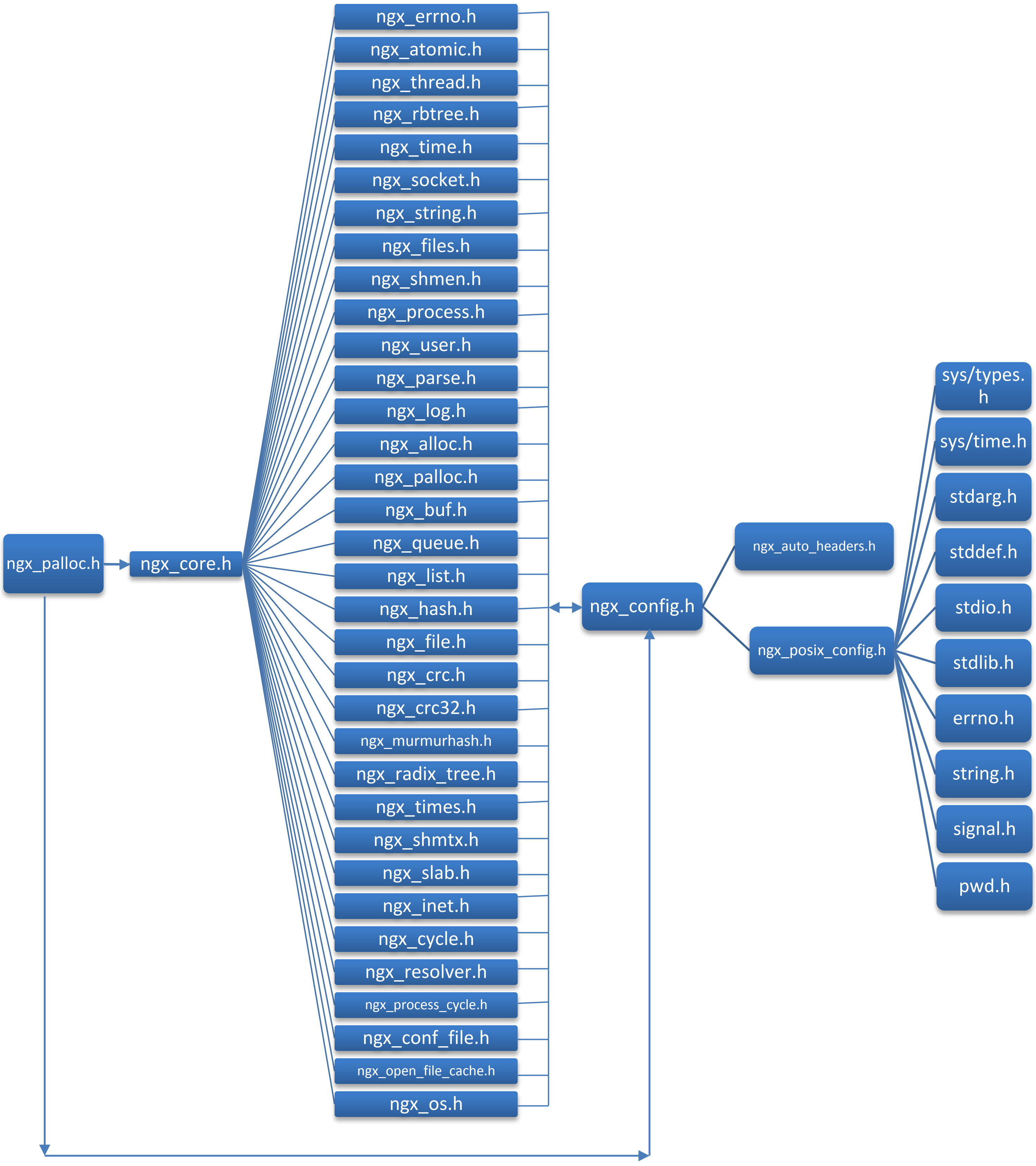
Data Structures Defined:

```
struct ngx_pool_cleanup_s {
    ngx_pool_cleanup_pt   handler;
    void                  *data;
    ngx_pool_cleanup_t    *next;
};

struct ngx_pool_large_s {
    ngx_pool_large_t     *next;
    void                 *alloc;
};


struct ngx_pool_s {
    ngx_pool_data_t       d;
    size_t                max;
    ngx_pool_t           *current;
    ngx_chain_t          *chain;
    ngx_pool_large_t     *large;
    ngx_pool_cleanup_t   *cleanup;
    ngx_log_t            *log;
}
```

Include Dependency Graph :


ngx_errno.h
ngx_atomic.h
ngx_thread.h
ngx_rbtree.h
ngx_time.h
ngx_socket.h
ngx_string.h
ngx_files.h
ngx_shmen.h
ngx_process.h
ngx_user.h
ngx_parse.h
ngx_log.h
ngx_alloc.h
ngx_palloc.h
ngx_buf.h
ngx_queue.h
ngx_list.h
ngx_hash.h
ngx_file.h
ngx_crc.h
ngx_crc32.h
ngx_murmurhash.h
ngx_radix_tree.h
ngx_times.h
ngx_shmtx.h
ngx_slab.h
ngx_inet.h
ngx_cycle.h
ngx_resolver.h
ngx_process_cycle.h
ngx_conf_file.h
ngx_open_file_cache.h
ngx_os.h
ngx_palloc.h
ngx_core.h
ngx_config.h
ngx_auto_headers.h
ngx_posix_config.h
sys/types.h
sys/time.h
stdarg.h
stddef.h
stdio.h
stdlib.h
errno.h
string.h
signal.h
pwd.h