ELEC 377 Lab 2 Documentation

Emma Chan – 20206641
Jake Moffat – 20212243

Problem Statement

The purpose of this lab is to create a system that simulates a distributed monitor. This will be done by creating multiple systems that send updates to a central monitor which then proceeds to calculate statistics and print reports. In this lab, a separate thread (of multiple running in parallel) will read the data from a file, representing each of the monitors.

Description of Lab Chain

This program begins by saving the executable name of the program in global for error messages.

Then the monitor threads are started and the monitor_threads array is used to pass parameters. Then once this main program finishes, all the threads are terminated.

The init_shared function allocates and initializes the three semaphores (access_stats, access_summary, mutex) as well as initializes the shared object to the starting state. Then the monitor_update_status_entry function is called by the monitor thread for each entry in the monitor status files. It updates the entry in shared memory with the data sent from the simulated remote machine. There is a delay before each call, which is based on the last field of each line. It updates the entry in shared memory with the data sent from the simulated remote reader protocol of reader-writer.

Then, the reader_thread function is called which reads and then calculates the summary for each new entry and takes the pointer to the shared memory segment. In this function, the pointer to the total entries and max entries that have been read should recalculate and update for every new entry. The data used in this function is then updated to the central monitor, then released and posted. The reader_thread function continues until there are no more updates left to calculate, update and post. This function also deals with the alarms shown to the user when systems are down.

The printer thread is then initialized, and the function is called. The printer_thread function uses the summary semaphore to gather data to display to the terminal. Upon completion of the printer_thread function, the program moves to pthread_exit(0) and return NULL, and ends to the program.

monitor_update_status_entry Thread:

This thread contains: the state of the machine thread (machine_state); the number of processes running in the machine (num_of_processes); the load factor of the machine (load_factor); the packets per second of the machine (packets_per_second); the discards per second of the machine (discards_per_second).

Reader Thread

This thread contains the number of machines (num_machines); the read update times (read_update_times); the read machines state (read_machines_state); the total processes (total_procs); the total packets per second across all machines (total_pps); the total discards per second across all machines (total_dps) and the total load factor of all machines (total_lf).

Printer Thread:

This thread contains: the print period (print_period); the number of machines (num_machines); the current up time (cur_uptime) and the current overall time (cur_time).