# main

E:\workspace\yh\OpenBridge-passos-proxy\open-falcon\src\

agent-DomeOS-new\main.go

**var** argPort = flag.Int("port", 8080, "port to listen")

startContainerMonitor

**func** startContainerMonitor() manager.Manager {  
 flag.Parse()  
  
 **if** \*versionFlag {  
 log.Printf("cAdvisor version %s (%s)\n", version.Info["version"], version.Info["revision"])  
 os.Exit(0)  
 }  
  
 setMaxProcs()  
  
 memoryStorage := memory.New(\*storageDuration, nil)  
  
 sysFs, err := sysfs.NewRealSysFs()  
 **if** err != nil {  
 log.Fatalf("Failed to create a system interface: %s", err)  
 }  
  
 containerManager, err := manager.New(memoryStorage, sysFs, \*maxHousekeepingInterval, \*allowDynamicHousekeeping, ignoreMetrics.MetricSet)  
 **if** err != nil {  
 log.Fatalf("Failed to create a Container Manager: %s", err)  
 }  
  
 mux := http.NewServeMux()  
  
 **if** \*enableProfiling {  
 mux.HandleFunc("/debug/pprof/", pprof.Index)  
 mux.HandleFunc("/debug/pprof/cmdline", pprof.Cmdline)  
 mux.HandleFunc("/debug/pprof/profile", pprof.Profile)  
 mux.HandleFunc("/debug/pprof/symbol", pprof.Symbol)  
 }  
  
 // Register all HTTP handlers.  
 err = cadvisorhttp.RegisterHandlers(mux, containerManager, \*httpAuthFile, \*httpAuthRealm, \*httpDigestFile, \*httpDigestRealm)  
 **if** err != nil {  
 log.Fatalf("Failed to register HTTP handlers: %v", err)  
 }  
  
 cadvisorhttp.RegisterPrometheusHandler(mux, containerManager, \*prometheusEndpoint, nil)  
  
 // Start the manager.  
 **if** err := containerManager.Start(); err != nil {  
 log.Fatalf("Failed to start container manager: %v", err)  
 }  
  
 // Install signal handler.  
 installSignalHandler(containerManager)  
  
 log.Printf("Starting cAdvisor version: %s-%s on port %d", version.Info["version"], version.Info["revision"], \*argPort)  
  
 addr := fmt.Sprintf("%s:%d", \*argIp, \*argPort)  
 **go** http.ListenAndServe(addr, mux)  
  
 **return** containerManager  
}