

LinuxSocials Hackathon

Linux+Containers+AI

BUILD A DASHBOARD FOR MANAGING AND MONITORING DOCKER CONTAINERS RUNNING ON LINUX SYSTEMS.

[Metricbeat System] Containers overview - Kibana - Mozilla Firefox

localhost:5601/app/kibana#/dashboard/CPU-slash-Memory-per-container?_g=(refreshInterval:(display:off,pause:!f,value:0),time(from:now-2d,to:now),@mode:kiosk,@page:0),fullScreenMode:!options,(darkTheme)!panels~&_e=

Dashboard / [Metricbeat System] Containers overview

System Navigation [Metricbeat System]

System Overview | Host Overview | Containers overview

Container CPU usage [Metricbeat System]

Container ID	Process name	CPU user	CPU quota	CPU throttling	CPU kernel
user.slice	Web Content	81,646,280,000,000	0	0	18,383,240,000,000
user.slice	firefox	81,646,280,000,000	0	0	18,383,240,000,000
user.slice	gnome-shell	81,646,280,000,000	0	0	18,383,240,000,000
user.slice	java	81,646,280,000,000	0	0	18,383,240,000,000
user.slice	slack	81,646,280,000,000	0	0	18,383,240,000,000
crashplan.service	java	2,878,700,000,000	0	0	189,080,000,000
init.scope	systemd	6,250,000,000	0	0	16,000,000,000

Container Memory stats [Metricbeat System]

Container ID	Process name	Usage	Max usage	Page faults	Pages in memory	Pages out of memory	Inactive files	# Major page faults	Failures	TCP buffers	Huge pages	Swap caches	Swap usage	Block I/O
user.slice	ibus-daemon	13,678GB	13,937GB	1,085,777,932	683,707,669	680,180,308	696,852MB	30,766.5	0	13,678GB	0B	10,248GB	207,373MB	1,798,675
user.slice	metricbeat	13,624GB	13,93GB	1,075,724,703,324	675,417,695,919	671,904,351,622	611,071MB	28,938,486	0	13,624GB	0B	10,16GB	169,974MB	1,750,805,459
user.slice	atom	13,671GB	13,935GB	1,072,533,100,952	675,666,145,981	672,140,776,019	625,196MB	29,107,808	0	13,671GB	0B	10,214GB	164,516MB	1,754,182,76
user.slice	less	13,528GB	13,930GB	1,072,798,723	675,030,249	671,542,734	688,504MB	29,209	0	13,528GB	0B	10,085GB	205,324MB	1,762,859
user.slice	node	13,741GB	13,937GB	1,069,623,156,667	672,993,070	669,449,272	737,432MB	28,793,333	0	13,741GB	0B	10,223GB	146,921MB	1,744,092,667
crashplan.service	java	946,43MB	2,135GB	408,965,123	2,301,133,862	2,123,022,377	135,784MB	514,823	0	946,43MB	0B	521,687MB	9,305MB	952,406,915
init.scope	systemd	7,203MB	11,996MB	63,000,667	26,991,333	26,331,333	256KB	5	0	7,203MB	0B	2,199MB	818,667KB	863

Container Block IO [Metricbeat System]

Container ID	Process name	Total	I/O
user.slice	ibus-daemon	44,069GB	1,798,675

THE DASHBOARD SHOULD PROVIDE AN OVERVIEW OF ALL CONTAINERS, THEIR STATUSES, RESOURCE USAGE, AND IT'S GRAPHICAL REPRESENTATION FOR REACH CONTAINERS' RESOURCE USAGE.



USERS SHOULD BE ABLE TO RUN, START, STOP, AND RESTART CONTAINERS

Filter...

☐ On Only

Image ↑	State	Names
ansible/ansible-runner:1.4.4	<input type="checkbox"/> Off	command_runner
hashicorp/terraform:0.12.18	<input type="checkbox"/> Off	command
localstack/localstack:0.10.6	<input checked="" type="checkbox"/> On	command
redis	<input type="checkbox"/> Off	reverent_h
redis	<input type="checkbox"/> Off	fervent_he
redis	<input type="checkbox"/> Off	competen
redis	<input type="checkbox"/> Off	serene_pt
redis	<input type="checkbox"/> Off	eloquent_
redis:latest	<input checked="" type="checkbox"/> On	laughing_j
redis:latest	<input type="checkbox"/> Off	busy_ber

**HINT: YOU COULD USE DOCKER SDKS OR APIS TO INTERACT
WITH DOCKER ENGINE AND CREATE THE DASHBOARD USING
WEB TECHNOLOGIES LIKE REACT.JS OR VUE.JS.**



WHEN YOU SEE A HINT

***BONUS TASK: GIVE A SECTION TO INPUT DOCKERFILES
AND IT WILL BE CORRECTED AND OPTIMISED BY GPT IF
THERE IS ANY ERROR/REDUNDANCY.***

