

chapter 19.3	acknowledgment packet :
chapter 2.8	ahead-of-time (AOT) compilation :
chapter 1.10	Amazon Elastic Compute Cloud (ec2) :
chapter 2.4	application program :
chapter 2.3	application programming interface (API) :
chapter 3.6	asynchronous :
chapter 4.6	asynchronous procedure call (APC) :
chapter 4.4	asynchronous threading :
chapter 3.2	background :
chapter 1.9	balanced binary search tree :
chapter 3.8	big-endian :
chapter 1.9	binary search tree :
chapter 1.9	binary tree :
chapter 1.2	bit :
chapter 1.9	bitmap:
chapter 4.5	block :
chapter 3.6	blocking :
chapter 1.2	bootstrap program :
chapter 2.10	bottleneck :
chapter 3.5	bounded buffer :
chapter 2.10	BPF compiler collection (BCC) :
chapter 19.3	broadcast :
chapter 2.3	bug :
chapter 1.2	bus :
chapter 1.2	byte :

chapter 2.3	C library (libc) :
chapter 1.5	cache management :
chapter 19.8	cache-consistency problem :
chapter 1.5	caching :
chapter 19.3	checksum :
chapter 3.2	children :
chapter 19.6	client :
chapter 2.3	client :
chapter 19.6	client interface :
chapter 1.10	client system :
chapter 19.6	client-server model :
chapter 18.3	clone :
chapter 18.3	cloud computing :
chapter 2.2	command interpreter :
chapter 2.1	command-line interface (CLI) :
chapter 3.7	communication port :
chapter 2.3	communications :
chapter 19.5	compression :
chapter 19.4	computation migration :
chapter 19.3	congestion control :
chapter 3.8	connection-oriented socket (TCP) :
chapter 3.8	connectionless socket (UDP) :
chapter 18.3	consolidation :
chapter 3.5	consumer :
chapter 18.5	containers :
chapter 3.2	context :
chapter 3.2	context switch :
chapter 1.1	control program :
chapter 1.3	core :
chapter 2.10	core dump :
chapter 3.2	CPU scheduler :
chapter 3.2	CPU-bound process :
chapter 2.10	crash :
chapter 2.10	crash dump :

chapter 19.4	data migration :
chapter 4.2	data parallelism :
chapter 3.1	data section :
chapter 19.3	datagram :
chapter 2.3	debugger :
chapter 2.10	debugging :
chapter 19.5	deduplication :
chapter 4.6	default signal handler :
chapter 19.8	delayed-write policy :
chapter 2.2	desktop :
chapter 1.2	device driver :
chapter 1.2	direct memory access (DMA) :
chapter 18.6	disk image :
chapter 19.7	diskless :
chapter 19.6	distributed file system (DFS) :
chapter 19.1	distributed system :
chapter 18.5	docker :
chapter 19.3	domain-name system (DNS) :
chapter 1.2	dynamic random-access memory (DRAM) :
chapter 2.5	dynamically linked libraries (DLLs) :
chapter 18.5	emulation :
chapter 18.1	emulator :
chapter 1.4	exception :
chapter 2.5	executable and linkable format (ELF) :
chapter 3.1	executable file :
chapter 19.5	fault-tolerant system :
chapter 1.5	file :
chapter 19.7	file migration :
chapter 1.10	file-server system :
chapter 1.10	firewall :
chapter 1.2	firmware :
chapter 19.3	flow control :
chapter 2.2	folder :
chapter 3.2	foreground :
chapter 1.2	gigabyte (GB) :
chapter 2.1	graphical user interface (GUI) :
chapter 19.1	graphics processing unit (GPU) :
chapter 18.1	guest :
chapter 1.7	guest :

chapter 19.6	Hadoop distributed file system (HDFS) :
chapter 1.2	hard disk drive (HDD) :
chapter 1.9	hash function :
chapter 1.9	hash map:
chapter 3.1	heap section :
chapter 19.5	heartbeat :
chapter 1.3	high-availability :
chapter 1.3	high-performance computing :
chapter 18.1	host :
chapter 19.3	host name :
chapter 19.3	host-id :
chapter 18.5	hypercall :
chapter 18.1	hypervisor :
chapter 1.5	I/O subsystem :
chapter 3.2	I/O-bound process :
chapter 19.2	Internet :
chapter 19.3	Internet protocol (IP) :
chapter 19.2	Internet Service Providers (ISPs) :
chapter 3.4	interprocess communication (IPC) :
chapter 1.2	interrupt :
chapter 1.2	interrupt priority level :
chapter 1.2	interrupt vector :
chapter 1.2	interrupt-handler routine :
chapter 18.5	Java virtual machine :
chapter 3.1	job :
chapter 1.1	kernel :
chapter 2.8	kernel abstractions :
chapter 1.4	kernel mode :
chapter 4.3	kernel threads :
chapter 1.2	kilobyte (KB) :
chapter 18.5	Kubernetes :
chapter 4.6	lightweight process (LWP) :
chapter 1.9	linked list :
chapter 2.5	linker :
chapter 1.9	list :
chapter 3.8	little-endian :
chapter 18.3	live migration :
chapter 19.1	load balancing :
chapter 2.8	loadable kernel module (LKM) :
chapter 2.5	loader :
chapter 19.2	local-area network (LAN) :
chapter 2.3	lock :
chapter 2.10	log file :
chapter 1.4	logical memory :

chapter 19.6	MapReduce :
chapter 1.2	maskable :
chapter 2.7	mechanism :
chapter 19.3	medium access control (MAC) address :
chapter 1.2	megabyte (MB) :
chapter 1.2	memory :
chapter 19.1	message :
chapter 3.7	message :
chapter 3.4	message passing :
chapter 2.3	message-passing model :
chapter 2.8	microkernel :
chapter 1.4	mode bit:
chapter 2.8	monolithic :
chapter 1.1	Moore's Law :
chapter 1.3	multicore :
chapter 1.3	multiprocessor :
chapter 1.4	multiprogramming :
chapter 1.4	multitasking :
chapter 4.1	multithreaded :
chapter 19.3	name server :
chapter 19.7	naming :
chapter 1.8	network :
chapter 18.6	network address translation :
chapter 1.10	network computer :
chapter 19.6	network file system (NFS) :
chapter 19.4	network operating system :
chapter 19.8	network virtual memory :
chapter 3.6	nonblocking :
chapter 1.2	nonmaskable interrupt :
chapter 1.2	nonvolatile memory (NVM) :
chapter 1.2	nonvolatile storage (NVS) :
chapter 19.3	OSI protocol stack :

chapter 1.3	parallelization :
chapter 3.2	parent :
chapter 1.2	petabyte (PB) :
chapter 18.6	physical-to-virtual (P-to-V) :
chapter 3.7	pipe :
chapter 1.10	platform as a service (PaaS) :
chapter 3.4	plug-in :
chapter 2.7	policy :
chapter 3.7	port :
chapter 3.8	port :
chapter 2.6	port :
chapter 19.3	port number :
chapter 1.4	privileged instructions :
chapter 1.4	privileged mode :
chapter 3.1	process :
chapter 2.3	process control :
chapter 3.1	process control block :
chapter 3.3	process identifier (pid) :
chapter 19.4	process migration :
chapter 2.3	process name :
chapter 3.2	process scheduler :
chapter 3.5	producer :
chapter 3.1	program counter :
chapter 18.1	programming-environment virtualization :
chapter 2.3	protection :
chapter 18.4	protection domain :
chapter 1.4	protection rings :
chapter 18.6	pseudo device driver :
chapter 4.4	Pthreads :
chapter 1.10	public cloud :
chapter 1.9	queue :
chapter 1.2	random-access memory (RAM) :
chapter 3.2	ready queue :
chapter 1.10	real-time operating systems (RTOS) :
chapter 2.5	relocatable object file :
chapter 2.5	relocation :
chapter 18.3	remote desktop protocol (RDP) :
chapter 19.4	remote file transfer :
chapter 1.5	resource manager :
chapter 19.1	resource sharing :
chapter 1.4	response time :
chapter 18.3	resume :
chapter 19.2	router :
chapter 2.3	run-time environment (RTE) :

chapter 19.5	scalability :
chapter 4.6	scheduler activation :
chapter 1.2	secondary storage :
chapter 1.6	security :
chapter 19.3	sequence number :
chapter 2.3	server :
chapter 3.2	service :
chapter 2.4	service :
chapter 19.4	session :
chapter 3.4	shared memory :
chapter 2.3	shared-memory model :
chapter 2.2	shell :
chapter 2.2	shell script :
chapter 4.6	signal :
chapter 2.3	single step :
chapter 4.1	single-threaded :
chapter 18.3	snapshot :
chapter 3.8	socket :
chapter 1.10	Software as a Service (SaaS) :
chapter 2.3	stack :
chapter 3.1	stack section :
chapter 3.1	state :
chapter 3.2	state restore :
chapter 3.2	state save :
chapter 1.3	storage-area network (SAN) :
chapter 3.8	stub :
chapter 2.4	subsystem :
chapter 1.4	supervisor mode :
chapter 18.3	suspend :
chapter 3.2	swapping :
chapter 1.3	symmetric multiprocessing (SMP) :
chapter 3.6	synchronous :
chapter 4.4	synchronous threading :
chapter 2.3	system call :
chapter 1.4	system mode :
chapter 2.4	system service :
chapter 2.3	system-call interface :

chapter 1.2	terabyte (TB) :
chapter 3.1	text section :
chapter 3.1	thread :
chapter 4.6	thread cancellation :
chapter 4.4	thread library :
chapter 4.5	thread pool :
chapter 4.6	thread-local storage (TLS) :
chapter 1.8	transmission control protocol/Internet protocol (TCP/IP) :
chapter 19.5	transparent :
chapter 2.8	trap :
chapter 18.4	trap-and-emulate :
chapter 3.3	tree :
chapter 18.1	type 0 hypervisor :
chapter 18.1	type 1 hypervisor :
chapter 18.1	type 2 hypervisor :
chapter 3.5	unbounded buffer :
chapter 19.3	user datagram protocol (UDP) :
chapter 2.1	user interface (UI) :
chapter 1.4	user mode :
chapter 3.1	user programs :
chapter 4.3	user thread :
chapter 4.6	user-defined signal handler :
chapter 18.4	virtual CPU (VCPU) :
chapter 1.7	virtual machine (VM) :
chapter 18.1	virtual machine manager (VMM) :
chapter 1.4	virtual memory :
chapter 18.6	virtual-to-physical (V-to-P) :
chapter 1.7	virtualization :
chapter 18.2	VMware :
chapter 1.2	volatile :
chapter 1.2	von Neumann architecture :
chapter 3.2	wait queue :
chapter 19.2	wide-area network (WAN) :
chapter 19.2	WiFi :
chapter 1.10	wireless network :
chapter 1.2	word :
chapter 19.2	World Wide Web (WWW) :
chapter 19.8	write-back caching :
chapter 19.8	write-on-close policy :
chapter 19.8	write-through policy :
chapter 3.3	zombie :