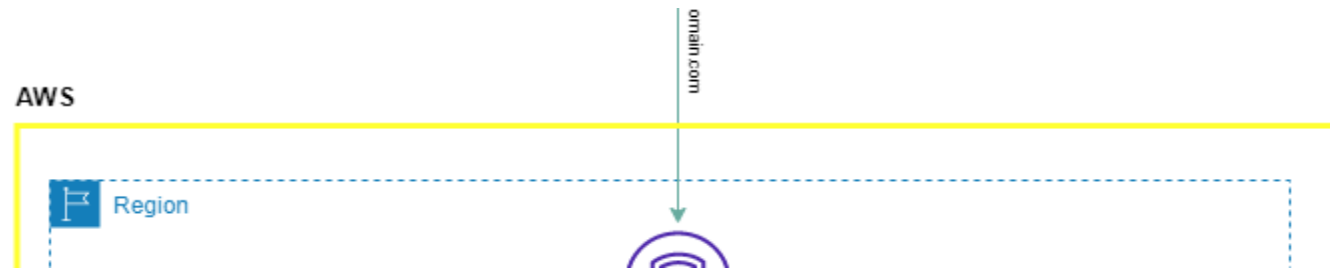
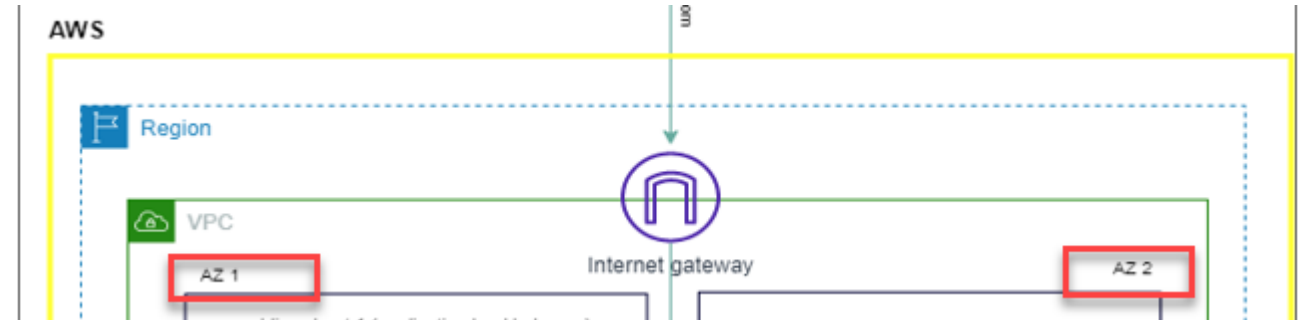
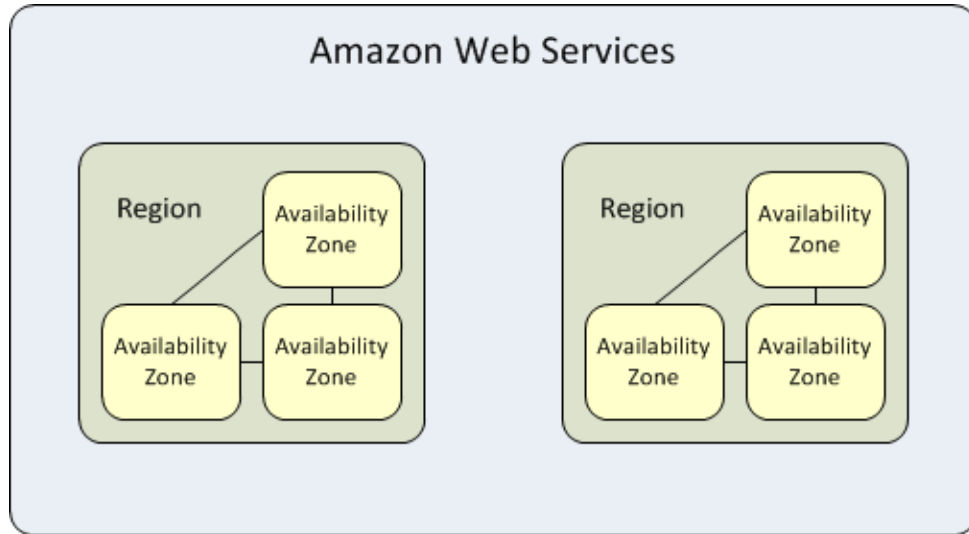


အရင်ဆုံး AWS region တွေအကြောင်း တီးခေါက်ကြတာပေါ့။ Multi-player online game တွေ ဘာတွေဆော့ရင် SEA server, Korea server တွေရှိတာပေါ့။ AWS မှာလည်း ဘယ် server မှာ ဆော့မှာလဲထက် ဘယ် region မှာ အလုပ်လုပ်မှာလည်း ရွေးရမှာပေါ့။ :3

Myanmar ကနေဆို မအူပင် (Mumbai xD) ရယ် Singapore region က နီးတယ် ပြော လို့ ရတယ်။ မြန်မာက production sites တွေက Singapore region မှာ ဆောက်တာများတယ်။ မအူပင်ကတော့ latency များတယ်ပေါ့ (အဲ့မှာ မစမ်းဖူးတော့ ကြားဖူးတာ ထည့်ထားပေးတာ ^^)။ ကျွန်တော်က တော့ lab testing တွေစမ်းရင် north Virginia region မှာစမ်းတယ်။ cheapest region in aws လို့ ပြောလို့ရတယ်။ aws services cost တွေက region ပေါ်မူတည်ပြီး ဈေးကွာတာ သိရမှာပေါ့။



Availability Zone (AZ) ဆိုတာကတော့ region တစ်ခုတည်းမှာ ရှိတဲ့ မတူတဲ့နေရာက server location တွေလို့ မှတ်လို့ရတယ်။ Singapore region (ap-southeast-1) မှာဆို ap-southeast-1a နဲ့ ap-southeast-1b ဆိုပြီး AZ 2ခု ရှိတယ်။



**Amazon Virtual Private Cloud (VPC)** က aws resources တွေ services တွေကို ထည့်ပြီး အလုပ်လုပ်တဲ့ virtual network ကြီးတစ်ခုပေါ့။

VPC က ပုံမှန်အပြင်က on premise network တွေလို data center နဲ့ သုံးသလိုဖြစ်အောင် ပုံဖော် simulateလုပ်ပေးတယ်။ cloud ပေါ်က ဆိုတော့ လိုသလို တိုးတာလျှော့တာ scale in scale out လုပ်လိုရတာ ကောင်းကွက်ပေါ့။

Aws console ကနေ ကိုယ်လိုချင်တဲ့ region ရွေးထားပြီး create VPC ဆောက်လို့ ရတယ်။ ipv4 CIDR block range သတ်မှတ်ပေးပြီး တန်းဆောက်လို့ရတယ်။

### Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

#### VPC settings

Resources to create Info

Create only the VPC resource or the VPC and other networking resources.

☒ VPC only

☐ VPC and more

Name tag - optional

Creates a tag with a key of 'Name' and a value that you specify.

IPv4 CIDR block Info

☒ IPv4 CIDR manual input

☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

IPv6 CIDR block Info

☒ No IPv6 CIDR block

☐ IPAM-allocated IPv6 CIDR block

☐ Amazon-provided IPv6 CIDR block

☐ IPv6 CIDR owned by me

Tenancy Info

#### Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

X

Value - optional

X

Remove

Add new tag

You can add 49 more tags.

Cancel

Create VPC

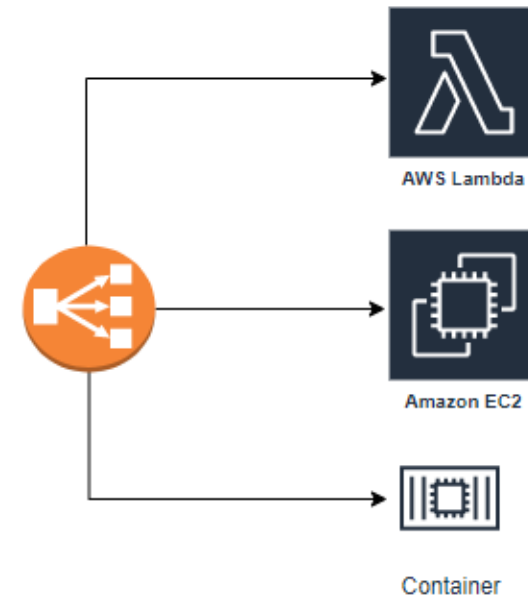
**Elastic load balancer (ELB)** က ဝင်လာတဲ့ inbound request ရဲ့ flow ကို manage လုပ်ပြီး ဆိုင်ရာ destination target group တွေဆီကို ပို့ပေးတာ လုပ်ပေးတာပေါ့။ target group တွေ ELB ရှိတဲ့ region ရဲ့ ကြိုက်တဲ့ different Availability Zone (AZ) မှာရှိလို့ရတယ်။

ဘာလို့ ELB ကို သုံးတာလဲ.. ဆိုပါစို့ web app တစ်ခု ကို Ec2 instance တစ်လုံးတည်း မှာ တင်ထားတယ်။ အကယ်၍ user request သိန်းချီဝင်လာပြီး သုံးမယ်ဆို traffic spike ဖြစ်ပြီး server fail သွားနိုင်တယ်။

ELB သုံးမယ်ဆို server အနည်းဆုံး ၂လုံးတော့ရှိရမှာပေါ့။ အဲ့တာမှ load balancing လုပ်လို့ရပြီး load balancer ဖြစ်မှာပေါ့။ server တစ်လုံး down သွားရင် နောက် တစ်လုံး redundant အနေနဲ့ ရှိအောင်လဲ သုံးကြတယ်။

ELB ကနေမှ သွားမဲ့ Target group တွေမှာ

Ec2 instances တွေ၊ lambda functions၊ IP address တွေရယ်အပြင် containers တွေပါ target group ထဲ ထည့်ထားလို့ရတယ် register လို့ရတယ် ။



## ELB မှာက Type 4 မျိုးရှိတယ်

### 1. Application load balancer (ALB)

သူ ကို အခု article diagram မှာသုံးပြထားတယ်ပေါ့။ OSI 7 layers မှာ application layer (layer 7) မှာရှိပြီး request level အနေနဲ့ အလုပ်လုပ်တယ်။ features တွေက advanced routing weight မျှပြီး route ခွဲတာတွေ၊ TLS termination SSL certificates တွေကို အလယ်ကနေ ထည့်ထားပြီး terminate လုပ်တာ၊ listen at different ports မတူတဲ့ port တွေမှာ request လို့ရအောင် support တယ်

### 2. Network load balancer (NLB)

Application ကို သုံး ရင် လုံးဝ low latency၊ အာထရ အာထရာ max performance ရချင်ရင် သုံးတယ်။ Connection level အနေနဲ့ အလုပ်လုပ်တယ်၊ သူလည်း target group တွေဆီ traffic route တာ support တယ်။ ultramax low latency high performance ဆိုတော့ million request per second ခံနိုင်တယ်။ :3

### 3. Gateway load balancer (GLB)

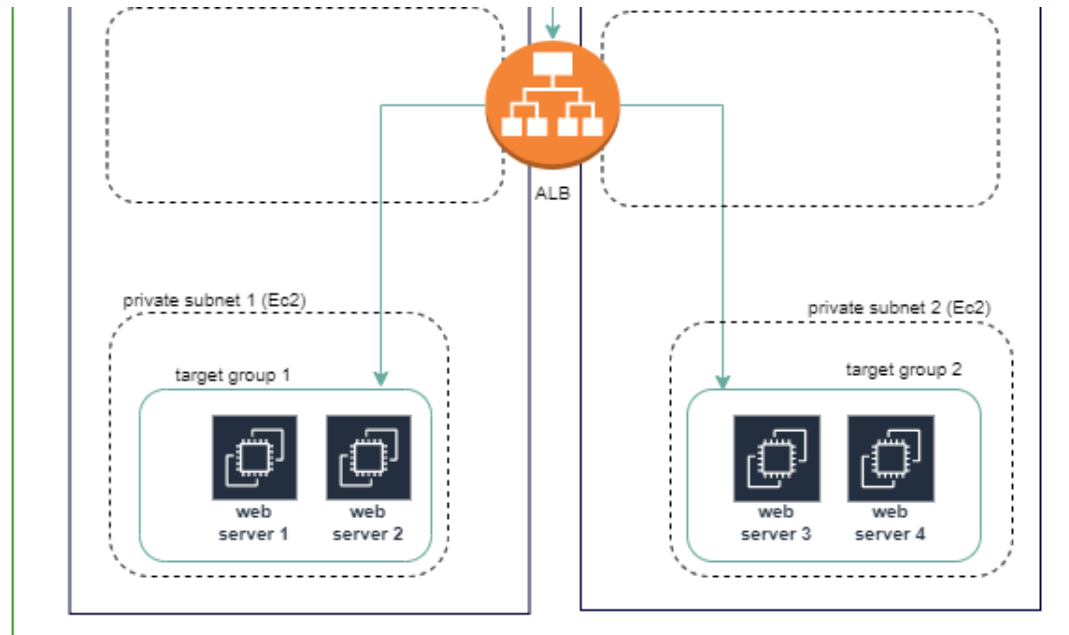
သူကကျ AWS ဆီက မဟုတ်ဘဲ အပြင်က vendor တွေရဲ့ virtual appliances (cloud ပေါ်ဆိုတော့ virtual ပဲပေါ့ :D) တွေကို deploy, scale, manage လုပ်ဖို့ဆိုသုံးတယ်။ third party virtual appliances တွေက ဘယ်ကရနိုင်လဲဆို AWS marketplace က ဝယ်လို့ရပါတယ်

### 4. Classic load balancer

ဒီကောင်က လာမဲ့ august 2022 မှာ aws က retire တော့မှာ( I think we can skip :3)

အောက်က ပုံမှာဆိုရင် **Application load balancer (ALB)** က လာသမျှ traffic request တွေရဲ့ဝင်ပေါက်လို့ အလုပ်လုပ်တယ်။ သူကနေမှ target group တွေဆီကို traffic distribute လုပ်ပေးတာပေါ့။

ပုံမှာဆို target group 1 ကို request hit ရင် server နှစ်လုံးရှိတာ တစ်လုံးfail နေရင် healthy ဖြစ်နေတဲ့ နောက်တစ်လုံးဆီ traffic ကို ELB က ရောက်အောင် လုပ်ပေးနိုင်တယ်။ ELB သုံးတာရဲ့ ကောင်းတာက managed by AWS ဖြစ်တာကြောင့် resilient and elastic ဖြစ်တာပေါ့။ 99.999999% fail ခဲပါတယ် :DD



**Security group** က Ec2 ဆီကို လာတဲ့ inbound traffic, ထွက်တဲ့ outbound traffic တွေကို control လုပ်ပေးတယ်။ Ec2 instances တွေအတွက်ဆို virtual firewall လေးပေါ့။ security group (sg) တွေက allow လုပ်လို့ပဲရတယ်၊ မလာပါနဲ့ဆိုပြီး deny လုပ်လို့မရဘူး။ ဒီမှာကျ network access control list(NACL) က allow ရော denyရောရတယ်။

okey... Ec2 instances တွေကို launch ရင် default security groupတွေတစ်ခါတည်းပါတယ်။ အဲ့တာကို မသုံးတာများပါတယ်။ security group အသစ်ဆောက် createပြီး inbound, outbound rule တွေပြင်လို့ရတယ်။ Ec2 instance ကို launch ပြီးမှ ဟိုportလေး ဖွင့်ချင်လို့ပါဆိုပြီးလည်း ကြိုက်တဲ့အချိန် edit inbound, outbound rules လာပြင်လို့ရတယ်။

Security group တွေက server တစ်ခုနဲ့တစ်ခု တူမှာ မဟုတ်ဘူး။ ec2 instanceတွေ က Same VPC ထဲမှာရှိနေရင်တောင် security group ချင်းထပ်တူကျမှာ မဟုတ်ဘူး။ ပြင်လို့ရတယ်။

### Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

#### Inbound rules Info

Security group rule ID	Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>	
sgr-07a08cdc33d4bc62	HTTPS	TCP	443	Custom		Delete
sgr-02b0a490435a3ab2b	SSH	TCP	22	My IP		Delete
sgr-0364f464187c2f862	HTTP	TCP	80	Custom		Delete

SSH access ကို ကိုယ်ရဲ့ IP ကနေမှ ဝင်ခိုင်းမယ်

http, https request ကို internet ကြိုက်တဲ့နေရာက ဝင်ခိုင်းမယ်

Ye Phone Myat

0.0.0.0/0 X

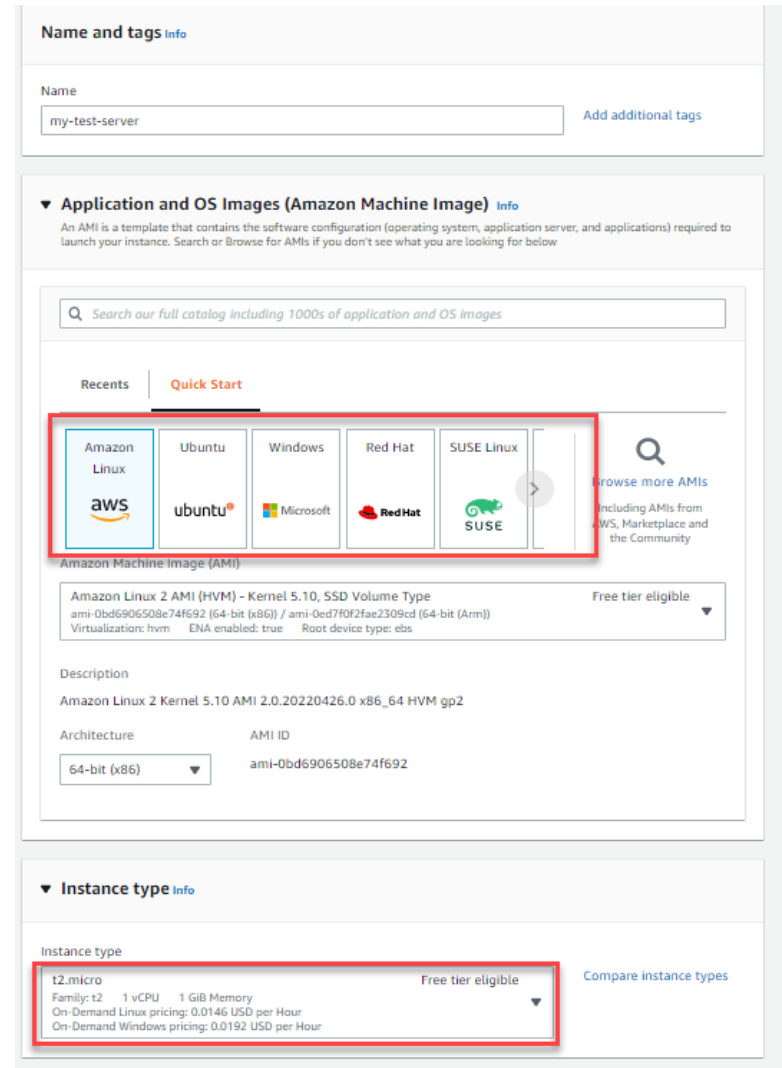
2/32 X

0.0.0.0/0 X

Amazon EC2... အရည်ကောက်ကတော့ Amazon Elastic Compute Cloud ပေါ့။ Compute cloud ဆိုတဲ့အတိုင်းပဲ compute workload အတွက် type အမျိုးအစားစုံစုံလင်လင်နဲ့ AWS console ပေါ်မှာတင်ရွေးလို့ရတယ်

အပြင်က on premise server တစ်လုံးကို cloud ပေါ်မှာ custom ဆင်သလိုပဲ။ Default အတိုင်း launch သုံးချင်လည်းရတယ်။ processor, storage, networking, OS ကြိုက်ရာ တိုးလျှော့လုပ်ပြီး clickရုံပဲ launch ဖို့။

Ec2 instance type ပေါ် မှုတည်ပြီး ဈေးလည်းကွာတယ်။ test ပြီးသုံးဖို့ဆို Free tier eligible ဖြစ်တဲ့ အလုံးတွေက 750 hours free ဆင်းလို့ရတယ် :DD eg. t2.micro





## EC2 Instance type တွေက

- General purpose
- Compute optimized
- Memory optimized
- Accelerated optimized
- Storage optimized ဆိုတဲ့အောက်မှာအမျိုးအစား ၅၀၀ကျော်ကွဲသွားတယ်။

**General purpose instances** တွေဆိုရင် web server အနေနဲ့ အသုံးများတယ်။ ပုံမှန် cpu ရော memory ရောမျှသုံးရတဲ့ workload တွေ သုံးသင့်တယ်။ T, M တွေနဲ့ စတယ် (eg. t2, m5)

**Compute optimized instances** တွေကတော့ computing power အရမ်းသုံးရတဲ့ workload တွေ အသုံးများတယ်။ media transcoding တို့ gaming servers တို့ modelling လို workloadတွေအတွက်ပေါ့။ C နဲ့စတယ်။ (eg. c4, c5)

**Memory Optimized instances** တွေက large scale data set တွေကို မြန်မြန်ဆန်ဆန် နဲ့ processing လုပ်နိုင်ဖို့သုံးတယ်။ R, X (R5, X1)တွေနဲ့စတယ်။ Storage optimized နဲ့က local storage ကနေ read, write ဖို့အတွက်လား performance processing အတွက်လားပေါ်မူတည်ပြီးusage ကွာသွားမယ်

**Accelerated optimized instances** တွေက hardware accelerators တွေ co-processor တွေပါတဲ့ instances တွေ။ data pattern တွေ numerical number တွေနဲ့ ဆိုင်တဲ့ intensive workload တွေအတွက်သုံးတယ်။ p4, g5, f1 etc တွေနဲ့စပြီး typeတွေကွဲတယ်။

**Storage optimized instances** တွေက local storage ကနေ high sequential read write လုပ်မဲ့ workload တွေသုံးတယ်။ Low latency, random IOs ရတယ်။ SSD, HDD storage ပေါ်မူတည်ပြီး i3, d3 ဆိုပြီးကွဲတယ်။ အသေးစိတ်က ရေးမကုန်လို့ ရှာဖတ် :DD

type တွေက များတယ် ဘာအသုံးများလဲဆိုတော့ general purpose ကတော့ ဈေး အသက်သာဆုံး :))  
ကျန်တာကတော့ usage workload ပေါ်မူတည်ပြီး သုံးကျတယ်

Type တွေ သိပြီဆိုတော့ instances တွေကို ဘယ်လို ဝယ်သုံးလို့ရလဲ။ **Purchasing plan** တွေပေါ့။

**On demand instances** ကြားဖူးတဲ့အတိုင်းပဲ ဒီကောင်က သုံးသလောက်ပေး၊ ကျသလောက်ရှင်း။ ၂၂၃လ အချိန်တို သုံးဖို့တို့ testing နဲ့ development အတွက်သင့်တော်တယ်

### **Reserved instances**

ကိုယ်က instance တစ်လုံးကို ကြာကြာသုံးမယ်ဆို on demand နဲ့က မကိုက်တော့ဘူး။ အချိန်တစ်ခုစာ သုံးမယ်သေချာရင် reserved instances တွေဝယ်လို့ရတယ်။ အချိန်က တော့ တစ်နှစ် or သုံးနှစ် ဆိုပြီးရှိတယ်။ seafood city hotpot ticket reserve ဝယ်သလိုပဲ လျှော့ဈေးdiscountရတယ် ဆိုင်ရောက်မှ ဝယ်တာထက် :D။ AWS reserved instances မှာတော့ အကုန်ကြိုပေးပြီး reserved plan ဝယ်စရာမလိုဘူး။

All upfront - အကုန်ကြိုပေး။ (Discount အများဆုံးရတယ်)

Partial upfront - နည်းနည်းကြိုရှင်း all front ထက် discount ပိုနည်းတယ်။

No upfront - အနည်းဆုံး discount ပဲရမယ်။ on demand ထက်တော့ သက်သာပါသေးတယ် ☹

## Scheduled instances

တစ်နှစ်စာဝယ်လို့ရတယ်။ နည်းနည်းဆန်းတာက ဘယ်အချိန်၊ ဘယ်ရက်တွေမှာ run ပါ့ မယ်ဆိုပြီး recurring ပေးထားလို့ရတယ်။ လတိုင်း လချုပ်စာရင်း ထုတ်တဲ့ရက်ပဲ runတဲ့ server တွေဆို ကိုက်တာပေါ့။

## Spot instances

AWS မှာ ပိုနေတဲ့ ec2 capacity ကို သုံးတာပြောလို့ရတယ်။ ကိုယ်လိုတဲ့ compute power ကို လေလံပစ် bid ပြီး ဝယ်သုံးရတယ်။ ဈေးချိုချိုနဲ့ compute power အများကြီး သုံးဖို့ဆိုရတယ်။ production အတွက် မသုံးသင့်ဘူး aws ပိုနေတဲ့ capacity ကို bid ပြီး သုံးတာဖြစ်တဲ့အတွက် ပြန်လိုလာပြီး spot instances တွေ interrupted ခံရနိုင်တယ်။

## Dedicated hosts

ကိုယ်ရဲ့ instances တွေကို run ဖို့ AWS ဆီကနေ physical host တစ်ခုလုံးငှါးပစ်လိုက်တာ။ aws physical host မှာ ကိုယ့်မှာ ရှိပြီးသား license တွေပြန်သုံးလို့ရတယ် သက်သာချင်ရင်။

## Dedicated instances

ဒီကောင်က အကောင်တစ်ခုအတွက် သီးသန့် hardware level ပါ ခွဲထားပေးပြီး သုံးလိုရတယ်။ dedicated specifically ဖြစ်တယ်ပေါ့။ hourlyပေးလိုရတယ်။

## Capacity reservations

ဘယ် AZ မှာ၊ ဘယ်လို specifications နဲ့ ဆိုပြီး ကြိုပြီး reserve လိုရတယ်။ သုံးသုံး မသုံးသုံးပေါ့။

## Saving plan (recommended for long term workload)

ဒီကောင်ကတော့ တစ်နှစ် or သုံးနှစ် မှာ တစ်နာရီကို ဘယ်လောက်သုံးပါမယ် commitment ပေးပြီး ဝယ်လိုရတယ်။ **reserved instances** နဲ့ ဘာကွာလဲဆို type ရွေးစရာမလိုတော့ဘူး။ တစ်နာရီကို ဘယ်လောက်သုံးပါမယ်ဆိုပြီးပေးထားတဲ့ commitment ဝင်ရင် discount ရတယ်။ any type, ဘာ instance type သုံးသုံး။