Networking

Virtual Private Cloud

VPC - Private networking on the cloud.
Subnets make up the VPC and are loosely related with Availability zones
Ip address are defined with (CIDR blocks) Classless Inter Domain Routing
VPC peering is used to interact with different VPCs.

Default VPC is made available by AWS in every region.
Default VPC should not be deleted (create a support ticket to AWS if needed to recover the default VPC)
Route table - Internet gateway - NAT Gateway -
CIDR - 0.0.0.0/prefix

Ex.1: x.x.x.x/24 => 32-24=8 =>2^8 = 256
Total 256 IPs are possible in this range.
So the IPs become - 192.168.1.(0-255)

Ex.2: x.x.x.x/20 => 32-20=8 =>2^8 = 256
Total 256 IPs are possible in this range.
So the IPs become - 192.168.1.(0-255)
So it can be divided into 4 times 64 IP addresses.
192.168.1.0-63
192.168.1.64-128
192.168.1.129-192
192.168.1.193-255

Always 5 IP addresses are taken by AWS for internal use and not given for the user to utilize.
.0 to .4 is reserved for AWS.
5 CIDR blocks per subnet.

Address format	Difference to last address	Mask	Addresses		Relative to class A, B, C	Restrictions on *a*, *b*, *c* and *d* (0..255 unless noted)	Typical use
			Decimal	2^n			
a.b.c.d / 32	+0.0.0.0	255.255.255.255	1	2^0	1/256 C		Host route
a.b.c.d / 31	+0.0.0.1	255.255.255.254	2	2^1	1/128 C	*d* = 0 ... (2*n*) ... 254	Point to point links (RFC 3021)
a.b.c.d / 30	+0.0.0.3	255.255.255.252	4	2^2	1/64 C	*d* = 0 ... (4*n*) ... 252	Point to point links (glue network)
a.b.c.d / 29	+0.0.0.7	255.255.255.248	8	2^3	1/32 C	*d* = 0 ... (8*n*) ... 248	Smallest multi-host network
a.b.c.d / 28	+0.0.0.15	255.255.255.240	16	2^4	1/16 C	*d* = 0 ... (16*n*) ... 240	Small LAN
a.b.c.d / 27	+0.0.0.31	255.255.255.224	32	2^5	⅛ C	*d* = 0 ... (32*n*) ... 224	
a.b.c.d / 26	+0.0.0.63	255.255.255.192	64	2^6	¼ C	*d* = 0, 64, 128, 192	
a.b.c.d / 25	+0.0.0.127	255.255.255.128	128	2^7	½ C	*d* = 0, 128	Large LAN
a.b.c.0 / 24	+0.0.0.255	255.255.255.0	256	2^8	1 C		
a.b.c.0 / 23	+0.0.1.255	255.255.254.0	512	2^9	2 C	*c* = 0 ... (2*n*) ... 254	
a.b.c.0 / 22	+0.0.3.255	255.255.252.0	1,024	2^{10}	4 C	*c* = 0 ... (4*n*) ... 252	
a.b.c.0 / 21	+0.0.7.255	255.255.248.0	2,048	2^{11}	8 C	*c* = 0 ... (8*n*) ... 248	Small ISP / large business
a.b.c.0 / 20	+0.0.15.255	255.255.240.0	4,096	2^{12}	16 C	*c* = 0 ... (16*n*) ... 240	
a.b.c.0 / 19	+0.0.31.255	255.255.224.0	8,192	2^{13}	32 C	*c* = 0 ... (32*n*) ... 224	ISP / large business
a.b.c.0 / 18	+0.0.63.255	255.255.192.0	16,384	2^{14}	64 C	*c* = 0, 64, 128, 192	
a.b.c.0 / 17	+0.0.127.255	255.255.128.0	32,768	2^{15}	128 C	*c* = 0, 128	
a.b.0.0 / 16	+0.0.255.255	255.255.0.0	65,536	2^{16}	256 C = B		
a.b.0.0 / 15	+0.1.255.255	255.254.0.0	131,072	2^{17}	2 B	*b* = 0 ... (2*n*) ... 254	
a.b.0.0 / 14	+0.3.255.255	255.252.0.0	262,144	2^{18}	4 B	*b* = 0 ... (4*n*) ... 252	
a.b.0.0 / 13	+0.7.255.255	255.248.0.0	524,288	2^{19}	8 B	*b* = 0 ... (8*n*) ... 248	
a.b.0.0 / 12	+0.15.255.255	255.240.0.0	1,048,576	2^{20}	16 B	*b* = 0 ... (16*n*) ... 240	
a.b.0.0 / 11	+0.31.255.255	255.224.0.0	2,097,152	2^{21}	32 B	*b* = 0 ... (32*n*) ... 224	
a.b.0.0 / 10	+0.63.255.255	255.192.0.0	4,194,304	2^{22}	64 B	*b* = 0, 64, 128, 192	
a.b.0.0 / 9	+0.127.255.255	255.128.0.0	8,388,608	2^{23}	128 B	*b* = 0, 128	
a.0.0.0 / 8	+0.255.255.255	255.0.0.0	16,777,216	2^{24}	256 B = A		Largest IANA block allocation
a.0.0.0 / 7	+1.255.255.255	254.0.0.0	33,554,432	2^{25}	2 A	*a* = 0 ... (2*n*) ... 254	
a.0.0.0 / 6	+3.255.255.255	252.0.0.0	67,108,864	2^{26}	4 A	*a* = 0 ... (4*n*) ... 252	
a.0.0.0 / 5	+7.255.255.255	248.0.0.0	134,217,728	2^{27}	8 A	*a* = 0 ... (8*n*) ... 248	
a.0.0.0 / 4	+15.255.255.255	240.0.0.0	268,435,456	2^{28}	16 A	*a* = 0 ... (16*n*) ... 240	
a.0.0.0 / 3	+31.255.255.255	224.0.0.0	536,870,912	2^{29}	32 A	*a* = 0 ... (32*n*) ... 224	
a.0.0.0 / 2	+63.255.255.255	192.0.0.0	1,073,741,824	2^{30}	64 A	*a* = 0, 64, 128, 192	
a.0.0.0 / 1	+127.255.255.255	128.0.0.0	2,147,483,648	2^{31}	128 A	*a* = 0, 128	
0.0.0.0 / 0	+255.255.255.255	0.0.0.0	4,294,967,296	2^{32}	256 A		

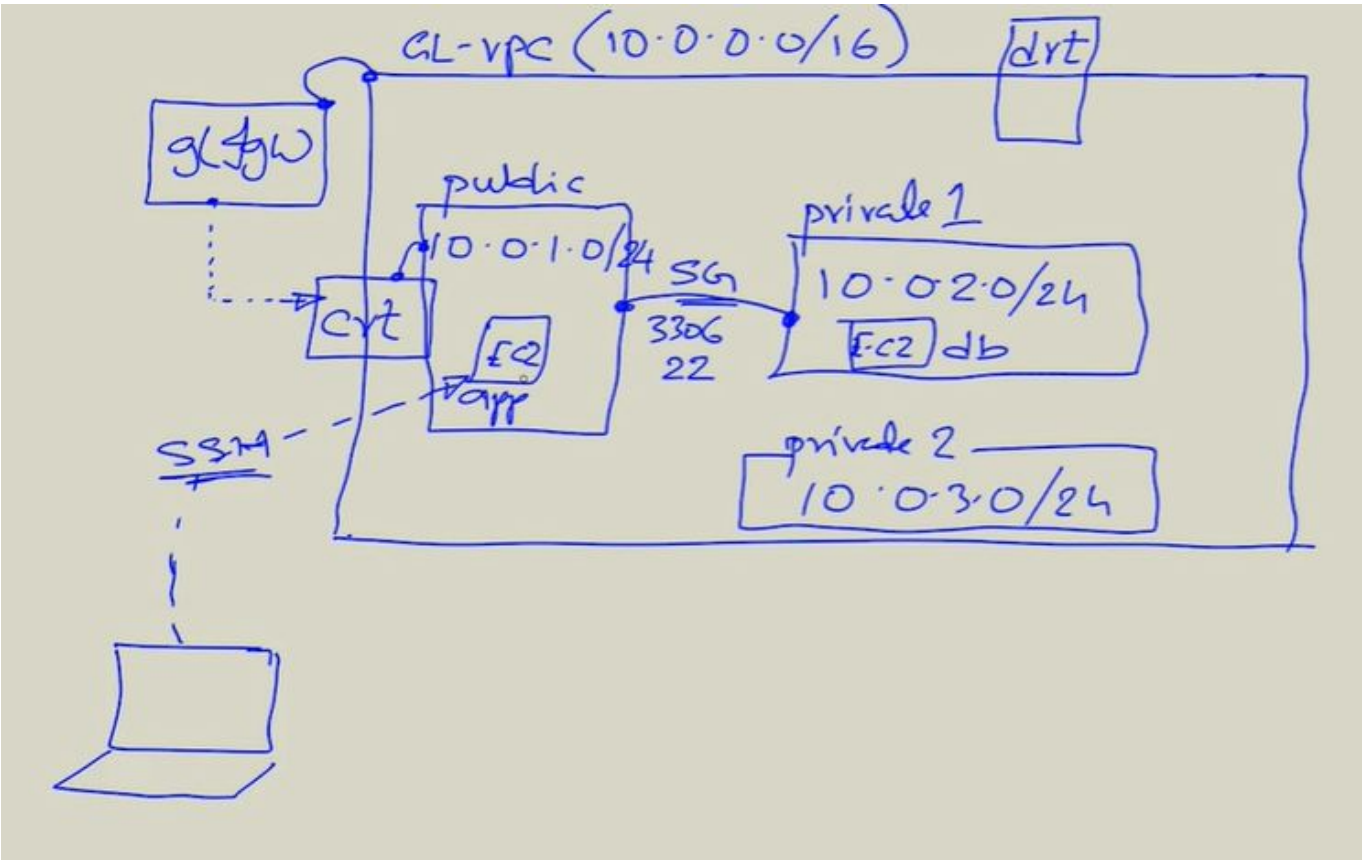
Connecting local subnets and the cloud VPC subnets is called Hybrid cloud.
This is done thru VPN connection or thru Amazon Direct connect.(Dedicated cables - so that connection does not go thru the internet)

If more data from your local needs to be transferred to AWS cloud then Snowball, snowmobile can be used.
Incremental data can be transferred thru the direct connect or VPN connection.

Security groups are stateful - once a port is opened for inbound connections the outbound is opened by default for the same port.
Incase of ACL, it is stateless and needs to be specified explicitly to allow the port inbound and outbound connections. Else it is blocked by default.

Amazon VPC limits
Custom limits of VPC per region is 5. (can increase with request to AWS support.)
Subnets 200 per VPC.
IPv4 ->CIDR 1+4 per VPC
IPv6 ->CIDR 1 per VPC

5 elastic IP per region (costs for unused address)
NACL 200
Rules per ACL 20
VPC FLow logs captures all the access and entry in and out of the



Each EC2 instance performs **source**/**destination** checks by default. This means that the instance must be the **source** or **destination** of any traffic it sends or receives. However, a NAT instance must be able to send and receive traffic when the **source** or **destination** is not itself.

Public Internet ⇒> Internet gateway ⇒> Default Route Table(or custom route table) ⇒> Route entry to public subnet ⇒> EC2 instance inside the public subnet ⇒> Private subnet(entry thru security group).

VPC peering request and acceptance has to happen for proper communication between the VPCs. Also the Route table entry for the VPC peering connection needs to be added on both the route tables of each VPCs to establish the connection.

VPC endpoints are available to handle the connection to AWS managed services such as DynamoDB, S3 etc to be securely connected to the VPC without letting the data flow thru internet.

Route 53

Direct and manage all traffic to your domain. Share the resources from different regions and load balance between the resources. Can register a domain with AWS or use 3rd party domains managed thru route 53.
Having the resources nearer to the users(using Sydney region for Australian users) is more appropriate than to have the server in Europe and letting the Australian user experience unnecessary latency delays.

Web Application Firewall

Configuration can be made to protect against bad users(hackers):

1. cross site scripting attack
2. SQL injection
3. Bad bots
4. Scanners
5. http flooding
6. IP address restriction
7. Attack protection

Select the region first
Web ACL name
CloudWatch metric name
Resource type.(ELB / CloudFront /API gateway)

Rule - Procedure/pattern match - Condition(set of conditions)
Rule groups - pre-prepared set of rules.
Priority of the rules used.

Rate-limiting to your resource. Ex. More than 100 requests within 5mins.