Exam consultation

Balázs Sonkoly, Felicián Németh, István Pelle, Balázs Fodor

2024-12-03

Course objective (recap)

Introduce

- "Cloud Native" paradigm
  - used for the development
  - (of modern network functions)
- operational methods in cloud environments

Software

- exploiting various services offered by clouds
- inherently optimized for clouds
- based on new development methods and design patterns

Cloud APIs

- programming interfaces
- make programmer's life easier
- enable rapid development

Platforms

- public cloud providers
  - Amazon AWS, Google Cloud Platform, Microsoft Azure
- open-source solutions
  - Kubernetes, OpenStack

Get understand

- toolsets, methods via concrete examples
  - AWS Academy
  - Kubernetes
- how these systems operate
  - new computing models
  - networking solutions

Together with

- Cloud Native Technologies Laboratory
- Cloud Network Service Programing in Go

Brief summary of the course

Basic concepts: Infrastructure + Software

Design for Cloud

- Cloud computing
  - enablers
    - virtualization (VMs, containers, networks)
  - service models
    - IaaS, PaaS, SaaS, CaaS, FaaS
  - public vs private
- Serverless
  - the cloud service provider automatically provisions, scales and manages the infrastructure required to run the code
  - developers can focus solely on writing and deploying their code
- Infrastructure: compute, storage, network
- Cloud native
  - design the application considering the capabilities and the services of the cloud
  - exploit the scale, elasticity, resiliency, and flexibility the cloud provides

As a result

- applications built from containers
  - following the microservices architecture
- managed by cloud platform(s)
  - which can start/stop/migrate/scale hundreds of containers
- while the continuous communication is ensured among the components
  - by software-defined networks (SDN)
- Microservices (architectural style)
  - application = collection of services
  - independently deployable
  - loosely coupled
- Design patterns
  - Publisher-Subscriber
  - Circuit Breaker
  - Gateway Aggregation
  - Sidecar
  - Externalized state, etc.

Infrastructure: Compute, Storage/Database, Networks

AWS Academy: Cloud Foundations

- Basics (CF 1-3)
- Security, IAM (CF 4, Lab 1)
- Compute (CF 6, Lab 3)
  - EC2, Lambda, Beanstalk
- Storage (CF 7, Lab 4)
  - EBS, S3, EFS
  - block, object, file storage
- Databases (CF 8, Lab 5)
  - RDS, DynamoDB
  - relational, NoSQL
- Networking (CF 5, Lab 2)
  - VPC
  - security groups, ACLs
- Autoscaling (CF 10, Lab 6)
  - monitoring, load balancing
- Cloud Architecture (CF 9)

Networking++

- Essential network functions
- NAT (Network Address Translation)
  - Source NAT (SNAT) / Masquerading
  - Destination NAT (DNAT), Port forwarding
  - `iptables`
- Firewall, packet filtering
  - stateless, stateful, application level
  - input / output / forward
  - `iptables` again, lot of examples...
- Edge Computing
  - resources: from cloud to edge
- 5G/6G Network exposure
  - "network API"
- SDN (Software Defined Networking)
  - softwarization of the control plane
- NFV (Network Function Virtualization)
  - softwarization of the data plane

Kubernetes: open-source, private cloud platform

Pod

- smallest deployable units of computing
- can contain one or more containers
  - shared storage and network resources
- specification for how to run the containers, e.g.:
  - environment variables
  - ports to expose
  - compute resources required by the containers

Deployment

- contains the configuration of a Pod
- contains additional information about the desired state of the Pod(s)
  - e.g.: number of replicas
- periodically compares the actual state with the desired state and updates the running Pods

Service

- an abstraction which helps to expose groups of Pods over a network
- one service usually targets Pods from one deployment
- the Pods can be accessed via one endpoint
- the Service automatically handles the load balancing among the targeted Pods

HPA (Horizontal Pod Autoscaler)

- it can be connected to a Deployment
- auto-scaling rules can be configured for the resource usage
  - if the actual resource exceeds the predefined threshold, HPA adjusts the number of Pods
- Autoscaling in general
  - horizontal/vertical, reactive/proactive,...

Developing with cloud APIs / Developing on AWS

AWS Academy: Cloud Developing

- AWS CLI, AWS SDK (CD 1-2,4 Lab 2.1)
  - CloudShell, Cloud9 (?), IAM
- Storage solutions (CD 3, Lab 3.1)
  - AWS S3
- NoSQL solutions (CD 5, Lab 5.1)
  - DynamoDB
- REST APIs (CD 6, Lab 6.1)
  - API Gateway
    - RESTful (HTTP/REST), WebSockets
- Lambda (CD 7, Lab 7.1)
  - event-driven, FaaS
- Function orchestration (CD 11, Lab 11.1)
  - Step Functions
- Messaging services (CD10, Lab 10.1)
  - SQS, SNS, Kinesis

Event-driven programming

- programming paradigm
- the program is determined by events such as
  - user actions
  - messages from other programs
- coming from an event source
- passing through a channel
- the program responds to events by executing predefined event handlers
- pull model vs push model
- related design patterns
  - publisher-subscriber, competing consumers, queue-based load leveling
- messaging services
  - queues, pub/sub messaging, streams
- debugging, monitoring
  - CloudWatch, X-Ray, Prometheus, ...

Exam: format and examples

Format, topics

Google Quiz

- Multi-choice test!
- each good answer is needed for the score

Theoretical questions

- Based on the lectures
- and practices
- (see the course summary)

Practice

- Exercises in an own environment
- topics: Kubernetes, network functions, AWS

AWS Academy materials

- Each module has a "Knowledge Check"
  - hopefully you have already realized that...
  - (required for accomplishing the gyaks)
- Lessons learned from the Labs
  - practice with an own Learner Lab environment
- Foundations course has a final Course Assessment
  - also required
- We can "borrow" questions from there
- but other (own) quiz questions will also appear

Frequently missed questions from the midterm (32/97)

How is Elastic Load Balancing (ELB) used with Amazon EC2 Auto Scaling? (Select TWO.)

- a) ELB performs health checks on new EC2 instances that are added to the Amazon EC2 Auto Scaling group.
- b) ELB distributes traffic between EC2 instances in an Auto Scaling group.
- c) ELB triggers an Auto Scaling event when a threshold is reached.
- d) ELB establishes the minimum and maximum number of instances in the Amazon EC2 Auto Scaling group.
- e) ELB automatically adds new instances to the Auto Scaling group when the load reaches a predetermined limit.

Frequently missed questions from the midterm (32/97)

How is Elastic Load Balancing (ELB) used with Amazon EC2 Auto Scaling? (Select TWO.)

⇒ a) ELB performs health checks on new EC2 instances that are added to the Amazon EC2 Auto Scaling group.

⇒ b) ELB distributes traffic between EC2 instances in an Auto Scaling group.

- c) ELB triggers an Auto Scaling event when a threshold is reached.
- d) ELB establishes the minimum and maximum number of instances in the Amazon EC2 Auto Scaling group.
- e) ELB automatically adds new instances to the Auto Scaling group when the load reaches a predetermined limit.

Frequently missed questions from the midterm (45/96)

What design patterns can be used to pass data between functions in the FaaS model? Choose the two best fitting!

- a) Externalized State
- b) Gateway aggregation
- c) Circuit Breaker
- d) Publisher-Subscriber

Frequently missed questions from the midterm (45/96)

What design patterns can be used to pass data between functions in the FaaS model? Choose the two best fitting!

⇒ a) Externalized State

- b) Gateway aggregation
- c) Circuit Breaker

⇒ d) Publisher-Subscriber

Frequently missed questions from the midterm (31/96)

Which of the following statements is/are true for the Kubernetes Service object?

- a) Service implements load balancing among the Pods it targets
- b) Manages the number of replicas of the Pods
- c) Automatically scales the Pods
- d) The Pods targeted by the Service can be access via one endpoint

Frequently missed questions from the midterm (31/96)

Which of the following statements is/are true for the Kubernetes Service object?

⇒ a) Service implements load balancing among the Pods it targets

- b) Manages the number of replicas of the Pods
- c) Automatically scales the Pods

⇒ d) The Pods targeted by the Service can be access via one endpoint

Frequently missed questions from the midterm (34/96)

Choose the true statements about Auto-Scaling!

- a) Horizontal scaling means adding or removing instances of a machine or container or runtime unit (function)
- b) Kubernetes' HPA uses proactive scaling by default
- c) Kubernetes' HPA uses a Rule-based method by default
- d) AWS Lambda uses CPU measurements to determine the number of replicas

Frequently missed questions from the midterm (34/96)

Choose the true statements about Auto-Scaling!

- ⇒ a) Horizontal scaling means adding or removing instances of a machine or container or runtime unit (function)
- b) Kubernetes' HPA uses proactive scaling by default
- ⇒ c) Kubernetes' HPA uses a Rule-based method by default
- d) AWS Lambda uses CPU measurements to determine the number of replicas

Frequently missed questions from the midterm (11/97)

What is/are the result(s) of the following command? Select the true statement(s)?

```
$ iptables -t nat -A PREROUTING -d 192.168.168.10 -p tcp --dport 2222 -j
DNAT --to-destination 10.0.0.153:22
```

- a) Setting up port forwarding, making all services of the machine with IP address 10.0.0.153 accessible from the outside
- b) Adding a new NAT rule to the nat table to expose TCP port 22 of an internal machine to an external network
- c) Adding a NAT rule to replace the destination IP address 192.168.168.10 for packets arriving on TCP port 2222
- d) Setting up port forwarding to make TCP and UDP port 22 of the machine with IP address 10.0.0.153 accessible from an external network

Frequently missed questions from the midterm (11/97)

What is/are the result(s) of the following command? Select the true statement(s)?

```
$ iptables -t nat -A PREROUTING -d 192.168.168.10 -p tcp --dport 2222 -j
DNAT --to-destination 10.0.0.153:22
```

- a) Setting up port forwarding, making all services of the machine with IP address 10.0.0.153 accessible from the outside

⇒ b) Adding a new NAT rule to the nat table to expose TCP port 22 of an internal machine to an external network

⇒ c) Adding a NAT rule to replace the destination IP address 192.168.168.10 for packets arriving on TCP port 2222

- d) Setting up port forwarding to make TCP and UDP port 22 of the machine with IP address 10.0.0.153 accessible from an external network

Frequently missed questions from the midterm (41/95)

What is/are the result(s) of the following command? Select the true statement(s)?

```
$ iptables -A FORWARD -d 10.0.0.0/24 -p tcp -m state ESTABLISHED,RELATED -j ACCEPT
```

- a) Enables established TCP traffic to and from hosts on the 10.0.0.0/24 network
- b) Permits established incoming TCP connections to the host from external networks
- c) Permits incoming and outgoing TCP traffic to/from the host if the connection is established
- d) Allows all TCP traffic to be forwarded

Frequently missed questions from the midterm (41/95)

What is/are the result(s) of the following command? Select the true statement(s)?

```
$ iptables -A FORWARD -d 10.0.0.0/24 -p tcp -m state ESTABLISHED,RELATED -j ACCEPT
```

⇒ a) Enables established TCP traffic to and from hosts on the 10.0.0.0/24 network

- b) Permits established incoming TCP connections to the host from external networks
- c) Permits incoming and outgoing TCP traffic to/from the host if the connection is established
- d) Allows all TCP traffic to be forwarded

Event-driven programming

When the event source directly invokes the event handler, and does not wait for the response, then:

- a) It is pull-based communication
- b) It is push-based communication
- c) It is asynchronous invocation
- d) The event is distributed using a queue

Event-driven programming

When the event source directly invokes the event handler, and does not wait for the response, then:

- a) It is pull-based communication
- ⇒ b) It is push-based communication
- ⇒ c) It is asynchronous invocation
- d) The event is distributed using a queue

Kubernetes Questions

1. We provide access to a pre-configured Kubernetes cluster with an AWS CloudFormation link
   - similarly to the 7^{th} lecture
   - should connect using the "EC2 Instance Connect browser-based client"
2. Questions
   - inquire about the state of the cluster
   - are solvable with `kubectl` commands
   - might need additional CLI arguments discussed in the practice session

- Example question: "How many services are in the default namespace?"

Kubernetes Questions

1. We provide access to a pre-configured Kubernetes cluster with an AWS CloudFormation link
   - similarly to the 7^{th} lecture
   - should connect using the "EC2 Instance Connect browser-based client"
2. Questions
   - inquire about the state of the cluster
   - are solvable with `kubectl` commands
   - might need additional CLI arguments discussed in the practice session

- Example question: "How many services are in the default namespace?"
- Solution:

```
$ kubectl get svc
NAME         TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.43.0.1      <none>        443/TCP   7m20s
my-service   ClusterIP   10.43.109.34   <none>        80/TCP    2m11s
$
```

⇒ Answer: 2

Another Kubernetes Question

- How many containers are running in the entire cluster (in every namespace)

Another Kubernetes Question

- How many containers are running in the entire cluster (in every namespace)
- Solution:

```
$ kubectl get pods -A
NAMESPACE     NAME                                      READY   STATUS      RESTARTS   AGE
default       my-server-859fcf67d-72j7m                 1/1     Running     0          2s
kube-system   coredns-7b98449c4-5f4x6                   1/1     Running     0          13m
kube-system   helm-install-traefik-crd-67dp2            0/1     Completed   0          13m
kube-system   helm-install-traefik-jx95k                0/1     Completed   1          13m
kube-system   local-path-provisioner-595dcfc56f-xzf7h   1/1     Running     0          13m
kube-system   metrics-server-cdcc87586-7kjnm            1/1     Running     0          13m
kube-system   svclb-traefik-f7a991d9-mcd46              2/2     Running     0          12m
kube-system   traefik-d7c9c5778-9xhkq                   1/1     Running     0          12m
```

⇒ Answer: 7

Yet Antoher Kubernetes Question

- How many pods are available to serve the service named "my-service"?

Yet Antoher Kubernetes Question

- How many pods are available to serve the service named "my-service"?
- Solution:

```
$ kubectl describe svc my-service
Name:              my-service
Namespace:         default
Labels:            <none>
Annotations:       <none>
Selector:          app=my-server
Type:              ClusterIP
IP Family Policy:  SingleStack
IP Families:       IPv4
IP:                10.43.109.34
IPs:               10.43.109.34
Port:              http  80/TCP
TargetPort:        8080/TCP
Endpoints:         10.42.0.10:8080,10.42.0.11:8080
Session Affinity:  None
Events:            <none>
```

⇒ Answer: 2 (Count the number of endpoints.)

Yet Antoher More Advanced Kubernetes Task

The DevOps team deployed their latest microservice to the Kubernetes cluster as a Deployment named exam-example in the default namespace. They want to run two instances of the Pods with the following resource requests 100 millicores CPU and 64 MB memory. But for some reason, the Pods are not starting. Why are the Pods associated with the deployment not starting? Fix the Deployment configuration!

Yet Antoher More Advanced Kubernetes Task

The DevOps team deployed their latest microservice to the Kubernetes cluster as a Deployment named exam-example in the default namespace. They want to run two instances of the Pods with the following resource requests 100 millicores CPU and 64 MB memory. But for some reason, the Pods are not starting. Why are the Pods associated with the deployment not starting? Fix the Deployment configuration!

- Possible Solution - Step 1:

```
$ kubectl get pods
  NAME                            READY   STATUS    RESTARTS   AGE
  exam-example-5f7c44968d-l5gx9   0/1     Pending   0          6m57s
  exam-example-5f7c44968d-qzxpt   0/1     Pending   0          6m57s
```

  - we can see that the Pods are indeed in Pending status

Yet Antoher More Advanced Kubernetes Task

- Possible Solution - Step 2:

```
$ kubectl describe deployment exam-example
Name:                   exam-example
...
Pod Template:
  Containers:
  nginx:
    Image:      docker.io/nginx:latest
    Port:       80/TCP
    Host Port:  0/TCP
    Limits:
      cpu:          100
      memory:       64Mi
   ...
Conditions:
  Type            Status  Reason
  ----            ------  ------
  Available       False   MinimumReplicasUnavailable
  Progressing     False   ProgressDeadlineExceeded
Events:
  Type    Reason             Age   From                   Message
  ----    ------             ----  ----                   -------
  Normal  ScalingReplicaSet  11m   deployment-controller  Scaled up replica set exam-example-5f7c44968d to 2
```

  - the Limits part looks a little strange but scheduling is based on resource Requests, not Limits, so let's check the Pods

Yet Antoher More Advanced Kubernetes Task

- Possible Solution - Step 3:

```
$ kubectl describe pod exam-example-5f7c44968d-l5gx9
Name:             exam-example-5f7c44968d-l5gx9
...
Containers:
  nginx:
    ...
    Requests:
      cpu:        100
      memory:     64Mi
    ...
Conditions:
  Type           Status
  PodScheduled   False
...
Events:
  Type     Reason            Age                From               Message
  ----     ------            ----               ----               -------
  Warning  FailedScheduling  15m                default-scheduler  0/1 nodes are available: 1 Insufficient cpu...
```

- It seems that it is a scheduling problem, there is not enough CPU, so let's check the CPU Requests configured!
- It is set to 100, which means 100 CPU cores. The task description said that the DevOps team wanted 100 millicores.

⇒ Answer: CPU request is set to 100 instead of 100m

AWS VPC/EC2 (1/2)

Given a preconfigured environment in the AWS Academy Learner Lab environment (deployed via CloudFormation based on the Neptun ID) answer the following questions (we might not ask all of them at the same time):

- What is the IPv4 CIDR block of the VPC?
- What Availability Zones does the VPC cover?
- How many public IP addresses are used by resources in this VPC?
- How many public/private subnets are in the given VPC?
- What are the IPv4 CIDR blocks of the subnets?
- In what Availability Zone is subnet `X`?
- What are the IP addresses/ports are allowed by the ACL of subnet `X`?
- Assume that there is an EC2 instance in a private subnet of the VPC, can it access the public internet? Analyze the setup of the VPC.
- What IP addresses/ports are allowed in the security group attached to EC2 instance `X`?
- Analyze the instance role of the EC2 instance `X`. Based on this, would it be able to access other AWS services in your account?

AWS VPC/EC2 (2/2)

Creating VPC/EC2 instances:

- Based on a given set of configuration parameters (depending on the Neptun ID) create a VPC in the AWS Academy Learner Lab environment.
- Start an EC2 instance with given specifications (instance type, AMI, security group, etc.) inside the VPC and run a given piece of code in it.
- Call an external grader script that
  - checks the correctness of your configuration
  - tries to access the code running inside the EC2 instance

AWS S3/Lambda/DynamoDB (1/2)

Given a preconfigured environment in the AWS Academy Learner Lab environment (deployed via CloudFormation based on the Neptun ID) answer the following questions (we might not ask all of them at the same time):

- Analyze the event log or resources generated by CloudFormation. What is the name of the S3 bucket/Lambda function/DynamoDB table that has been created for you?
- Analyze the S3 bucket. Is content inside the bucket publicly available?
- Analyze the Lambda function. What is the used runtime/memory/timeout configuration?
- Analyze the DynamoDB table. What are the configured RCU/WCU values? What is/are the attribute/attributes of the primary key? Is the primary key simple or composite? Are there any secondary indexes, if so what are their names?
- Uploading a file in the S3 bucket triggers the Lambda function that writes some data in the DynamoDB table. Upload a given file to the S3 bucket using a given method. A new item is added to the table. Give the item using the following format: `<attribute1>=<value1>,<attribute2>=<value2>,...`

AWS S3/Lambda/DynamoDB (2/2)

Creating S3 buckets/Lambda functions/DynamoDB tables in the AWS Academy Learner Lab environment:

- Create an S3 bucket with a given name.
- Create a Lambda function with a given name, using the programming language that you are familiar with, additional configuration parameters are provided based on the Neptun ID.
- Create a DynamoDB table using the configuration parameters that are provided based on the Neptun ID.
- Modify the Lambda function according to the following:
  - It is triggered by a `.csv` file upload in the S3 bucket.
  - The function reads the contents of the `.csv` file and inserts the items in the DynamoDB table.
- Run an external evaluator script that uploads a file in the S3 bucket and validates the contents of the DynamoDB table.
  - A sample `.csv` file is provided for testing
    - It has two lines: a header (with attribute names) and values
    - It has two "columns": the primary key and an attribute
    - Part of the content of the `.csv` used for the validation is different, but if the setup works with the test file, it should work with the validation file as well

AWS Step Functions

Given a preconfigured environment in the AWS Academy Learner Lab environment (deployed via CloudFormation based on the Neptun ID) answer the following questions (we might not ask all of them at the same time):

- How many steps are in the Step Function workflow?
- From start to end, give the types of the steps as a comma separated list.
- Given a specific test input what is the output?
- Analyze the Lambda functions making up the workflow. Modify the workflow to give a different output based on a given specification.
  - Run an external evaluator script that validates your modifications.