

Homework_Lesson31_Report

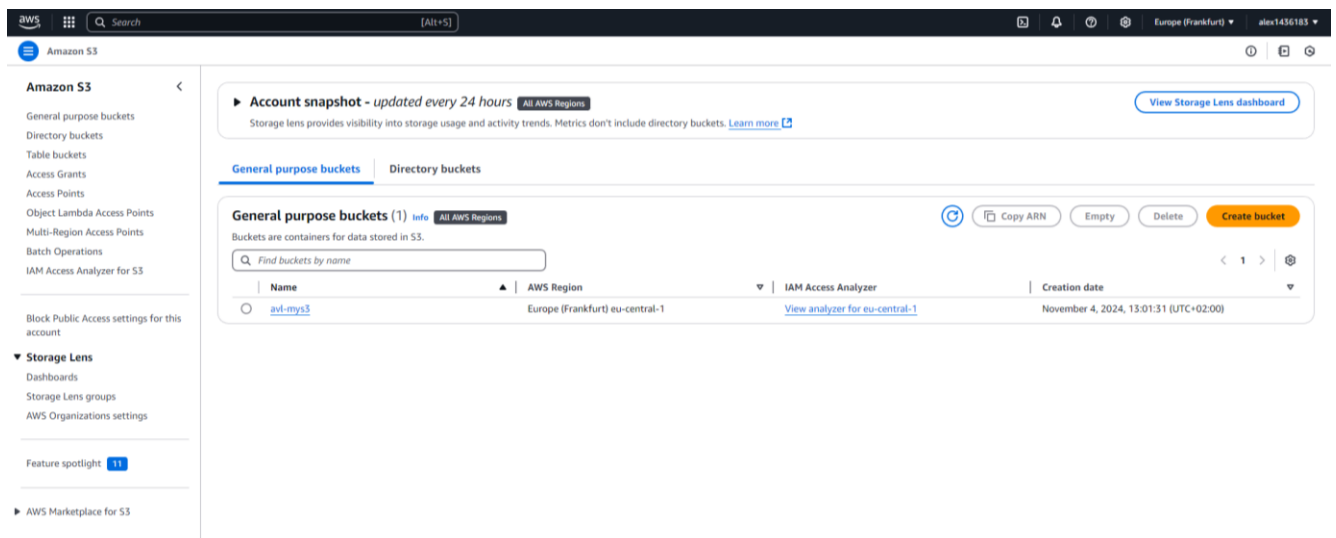
Задание:

"1. Создайте новый бакет Amazon S3/GCP Cloud Storage тремя способами: через GUI (консоль в браузере), с помощью CLI, конфиг в terraform.
2. Создайте Amazon EC2/GCP Compute Engine и настройте масштабирование, чтобы автоматически добавлять новые экземпляры в случае увеличения нагрузки. Сделайте это тремя способами: через GUI (консоль в браузере) с помощью CLI, конфиг в terraform.
3.* Создайте базу через GUI (консоль в браузере), с помощью CLI, конфиг в terraform данных Amazon RDS/GCP CloudSQL и подключитесь к ней из виртуальной машины, которую вы создали ранее. Сделайте это тремя способами: через GUI (консоль в браузере) с помощью CLI, конфиг в terraform."

Задание 1

Создайте новый бакета через GUI.

Заходим в AWS S3 и нажмем Create bucket.



Заполняем имя бакета и необходимые нам параметры и нажимаем Create bucket.

Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region
Europe (Frankfurt) eu-central-1

Bucket name Info
gui-mys3

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - optional
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Format: s3://bucket/prefix

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

☒ **ACLs disabled (recommended)**
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

☐ **ACLs enabled**
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership
Bucket owner enforced

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

☒ **Block all public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

Новый бакет создан.

> Buckets

Successfully created bucket "gui-mys3"

To upload files and folders, or to configure additional bucket settings, choose [View details](#).

View details

Account snapshot - updated every 24 hours

All AWS Regions

View Storage Lens dashboard

Storage lens provides visibility into storage usage and activity trends. Metrics don't include directory buckets. [Learn more](#)

General purpose buckets

Directory buckets

General purpose buckets (2)

Info

All AWS Regions

Refresh

Copy ARN

Empty

Delete

Create bucket

Buckets are containers for data stored in S3.

Find buckets by name

Name	AWS Region	IAM Access Analyzer	Creation date
avl-mys3	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	November 4, 2024, 13:01:31 (UTC+02:00)
gui-mys3	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	February 27, 2025, 10:01:38 (UTC+02:00)

Создадим через CLI
CLI у нас уже установлен и сконфигурирован.

```
C:\Windows\system32>aws s3api create-bucket --bucket cli-nys3 --create-bucket-configuration LocationConstraint=eu-central-1
{
  "Location": "http://cli-nys3.s3.amazonaws.com/"
}
```

Новый бакет создан.

General purpose buckets (3)

Info

All AWS Regions

Refresh

Copy ARN

Empty

Delete

Create bucket

Buckets are containers for data stored in S3.

Find buckets by name

Name	AWS Region	IAM Access Analyzer	Creation date
avl-mys3	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	November 4, 2024, 13:01:31 (UTC+02:00)
cli-mys3	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	February 27, 2025, 11:49:04 (UTC+02:00)
gui-mys3	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	February 27, 2025, 10:01:38 (UTC+02:00)

Создаем с помощью терформа.
Конфиг тераформа прикреплен отдельно.

```
PS D:\Terraform> terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create

Terraform will perform the following actions:

# aws_s3_bucket.my_bucket will be created
+ resource "aws_s3_bucket" "my_bucket" {
  + acceleration_status      = (known after apply)
  + acl                      = (known after apply)
  + arn                      = (known after apply)
  + bucket                   = "terraform-mys3"
  + bucket_domain_name      = (known after apply)
  + bucket_prefix            = (known after apply)
  + bucket_regional_domain_name = (known after apply)
  + force_destroy            = false
  + hosted_zone_id           = (known after apply)
  + id                      = (known after apply)
  + object_lock_enabled      = (known after apply)
  + policy                   = (known after apply)
  + region                   = (known after apply)
  + request_payer            = (known after apply)
  + tags_all                 = (known after apply)
  + website_domain           = (known after apply)
  + website_endpoint         = (known after apply)

  + cors_rule (known after apply)

  + grant (known after apply)

  + lifecycle_rule (known after apply)

  + logging (known after apply)

  + object_lock_configuration (known after apply)

  + replication_configuration (known after apply)

  + server_side_encryption_configuration (known after apply)

  + versioning (known after apply)

  + website (known after apply)
}

Plan: 1 to add, 0 to change, 0 to destroy.
```

Новый бакет создан.

General purpose buckets (4) Info All AWS Regions

🔄

📄 Copy ARN

Empty

Delete

Create bucket

🔍 Find buckets by name

< 1 > ⚙️

Name	AWS Region	IAM Access Analyzer	Creation date
<input type="radio"/> avl-mys3	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	November 4, 2024, 13:01:31 (UTC+02:00)
<input type="radio"/> cli-mys3	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	February 27, 2025, 11:49:04 (UTC+02:00)
<input type="radio"/> gui-mys3	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	February 27, 2025, 10:01:38 (UTC+02:00)
<input type="radio"/> terraform-mys3	Europe (Frankfurt) eu-central-1	View analyzer for eu-central-1	February 27, 2025, 14:58:17 (UTC+02:00)

Задание 2

Инстенс на EC 2 у созданный через GUI уже есть. Зайдем в него и нажеем Actions > Image and Templates > Create Image и создадим image нашей машины.

Create image [Info](#)

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.


Instance ID
 i-0a393c52055d03d07 (aws-ubuntu)

Image name

Maximum 127 characters. Can't be modified after creation.

Image description - optional

Maximum 255 characters

☒ Reboot instance
When selected, Amazon EC2 reboots the instance so that data is at rest when snapshots of the attached volumes are taken. This ensures data consistency.

Instance volumes

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev/s...	Create new snapshot from v...	8	EBS General Purpose SSD ~ ...	3000		<input checked="" type="checkbox"/> Enable	<input type="checkbox"/> Enable

Add volume

During the image creation process, Amazon EC2 creates a snapshot of each of the above volumes.

Tags - optional
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.

☒ Tag image and snapshots together
Tag the image and the snapshots with the same tag.

☐ Tag image and snapshots separately
Tag the image and the snapshots with different tags.

Создаем Auto Scaling group.

[EC2](#) > [Auto Scaling groups](#) > Create Auto Scaling group

- Step 1
Choose launch template
- Step 2
Choose instance launch options
- Step 3 - optional
Integrate with other services
- Step 4 - optional
Configure group size and scaling
- Step 5 - optional
Add notifications
- Step 6 - optional
Add tags
- Step 7
Review

Choose launch template [Info](#)

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name

Auto Scaling group name
Enter a name to identify the group.

Must be unique to this account in the current Region and no more than 255 characters.

Launch template [Info](#)

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

Select a launch template

Create a launch template [↗](#)

Cancel

Next

Создаем Launch template если у нас нету подходящего ранее созданного.

EC2 > Launch templates > Create launch template

Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - *required*

avl-lt

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.

Template version description

A prod webserver for MyApp

Max 255 chars

Auto Scaling guidance | Info

Select this if you intend to use this template with EC2 Auto Scaling

☒ Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

► Template tags

► Source template

Launch template contents

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

▼ Application and OS Images (Amazon Machine Image) - required Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon

macOS

Ubuntu

Windows

Red Hat

SUSE Linux

Debian

Q

По итогу создания Auto Scaling group у нас запустилось 2 истенса так как мы указали при создании Desired capacity 2.

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
<input type="checkbox"/>		i-0a698e02f5bfea476	Running	t2.micro	Initializing	View alarms +	eu-central-1a	ec2-18-184-194-30.eu-...	18.184.194.30	-
<input type="checkbox"/>	aws-ubuntu	i-0a393c52055d03d07	Running	t2.micro	2/2 checks passed	View alarms +	eu-central-1b	ec2-18-185-69-247.eu-...	18.185.69.247	-
<input type="checkbox"/>	ubuntu	i-0c7736f852ba260269	Stopped	t2.micro	-	View alarms +	eu-central-1b	-	-	-
<input type="checkbox"/>		i-0710b07d2f40b890e	Running	t2.micro	Initializing	View alarms +	eu-central-1b	ec2-18-199-173-74.eu-...	18.199.173.74	-

Создаем с помощью CLI Интстанс.

```
PS D:\Terraform> aws ec2 run-instances --image-id ami-0084a47cc718c111a --count 1 --instance-type t2.micro --key-name aws-1 --security-group-ids sg-0c0035be96ba711b7 --subnet-id subnet-0e8f515b5b7d37744 --tag-specifications 'ResourceType=instance,Tags=[{Key=Name,Value=av1_ubuntu}]'
```

```
{
  "ReservationId": "r-0dc65d81c690811b1",
  "OwnerId": "529088282556",
  "Groups": [],
  "Instances": [
    {
      "Architecture": "x86_64",
      "BlockDeviceMappings": [],
      "ClientToken": "af8f6a76-61c7-48e7-9bff-284bcalf7a8b",
      "EbsOptimized": false,
      "EnaSupport": true,
      "Hypervisor": "xen",
      "NetworkInterfaces": [
        {
          "Attachment": {
            "AttachTime": "2025-02-28T10:41:59+00:00",
            "AttachmentId": "eni-attach-0cfbf250a36d83ffe",
            "DeleteOnTermination": true,
            "DeviceIndex": 0,
            "Status": "attaching",
            "NetworkCardIndex": 0
          },
          "Description": "",
          "Groups": [
            {
              "GroupId": "sg-0c0035be96ba711b7",
              "GroupName": "launch-wizard-1"
            }
          ],
          "Ipv6Addresses": [],
          "MacAddress": "06:17:dc:28:53:1f",
          "NetworkInterfaceId": "eni-0facc704b3c99055",
          "OwnerId": "529088282556",
          "PrivateDnsName": "ip-172-31-44-183.eu-central-1.compute.internal",
          "PrivateIpAddress": "172.31.44.183",
          "PrivateIpAddresses": [
            {
              "Primary": true,
              "PrivateDnsName": "ip-172-31-44-183.eu-central-1.compute.internal",
              "PrivateIpAddress": "172.31.44.183"
            }
          ],
          "SourceDestCheck": true,
          "Status": "in-use",
          "SubnetId": "subnet-0e8f515b5b7d37744",
          "VpcId": "vpc-010a90ca45520da50"
        }
      ]
    }
  ]
}
```

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

Successfully initiated termination (deletion) of i-0d2a04c5802387330

Instances (4) Info

Last updated less than a minute ago

Connect

Instance state

Actions

Launch instances

Find Instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
<input type="checkbox"/>	aws-ubuntu	i-0a393c52055d03d07	Running	t2.micro	2/2 checks passed	View alarms +	eu-central-1b	ec2-18-1
<input type="checkbox"/>	av1_ubuntu	i-0d2a04c5802387330	Shutting-down...	t2.micro	2/2 checks passed	View alarms +	eu-central-1b	ec2-3-12
<input type="checkbox"/>	ubuntu	i-0c7236852ba260242	Stopped	t2.micro	-	View alarms +	eu-central-1b	-
<input type="checkbox"/>	av1_ubuntu1	i-0bc6d5039555a5601	Running	t2.micro	2/2 checks passed	View alarms +	eu-central-1b	ec2-35-1

Select an instance

Создаем Image на основе созданного Instance.

```
PS D:\Terraform> aws ec2 create-image --instance-id i-0bc6d5039555a5601 --name "my_ubuntu_av1" --no-reboot
```

```
{
  "ImageId": "ami-0d66b2a42da542211"
}
```

Amazon Machine Images (AMIs) (3) Info

Owned by me

Find AMI by attribute or tag

Recycle Bin

EC2 Image Builder

Actions

Launch instance from AMI

	Name	AMI name	AMI ID	Source	Owner	Visibility
<input type="checkbox"/>	MyUbuntu		ami-03273f79077391999	529088282556/MyUbuntu	529088282556	Private
<input type="checkbox"/>	my_ubuntu_av1		ami-0d66b2a42da542211	529088282556/my_ubuntu_av1	529088282556	Private
<input type="checkbox"/>	My_Ubuntu		ami-08516428035afad40	529088282556/My_Ubuntu	529088282556	Private

Select an AMI

Создаём Auto Scaling Group. Пропустил скрин создания в CLI Launch Template.

```
PS D:\Terraform> aws autoscaling create-auto-scaling-group --auto-scaling-group-name ubuntu-asg `
>> --launch-template "LaunchTemplateName=my-ubuntu-tem,Version=1" `
>> --min-size 1 --max-size 2 --desired-capacity 2 `
>> --vpc-zone-identifier "subnet-03207db2b91a2a05e,subnet-00669df1cc78cf133"
```

Настраиваем автоматическое масштабирование.

```
PS D:\Terraform> aws autoscaling put-scaling-policy --auto-scaling-group-name ubuntu-asg `
>> --policy-name scale-out-policy --adjustment-type ChangeInCapacity `
>> --scaling-adjustment 1 --cooldown 300
{
  "PolicyARN": "arn:aws:autoscaling:eu-central-1:529088282556:scalingPolicy:14f3b776-e8e5-4e34-9c5e-3fd9373eb8c2:autoScalingGroupName/
  "Alarms": []
}
```

Instances (5) <small>Info</small>							
Last updated 6 minutes ago Connect Instance state Actions Launch instances							
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/> All states							
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	aws-ubuntu	i-0a393c52055d03d07	Running	t2.micro	2/2 checks passed	View alarms	eu-central-1b
<input type="checkbox"/>	AutoScaling_Ubuntu	i-002109bb27e59cb06	Running	t2.micro	Initializing	View alarms	eu-central-1a
<input type="checkbox"/>	ubuntu	i-0c7236852ba260242	Stopped	t2.micro	-	View alarms	eu-central-1b
<input type="checkbox"/>	av1_ubuntu1	i-0bc6d5039555a5601	Running	t2.micro	2/2 checks passed	View alarms	eu-central-1b
<input type="checkbox"/>	AutoScaling_Ubuntu	i-097a8ff2335cf3494	Running	t2.micro	Initializing	View alarms	eu-central-1c

Select an instance

Приступим к выполнению в терраформе.

Конфиг тераформа прикреплен отдельно.

```
PS D:\Terraform\TEC2> terraform apply
aws_instance.av1_ubuntu1: Refreshing state... [id=i-0f6e10588573c812a]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create
/+ destroy and then create replacement

Terraform will perform the following actions:

# aws_ami_from_instance.my_ubuntu_1 will be created
+ resource "aws_ami_from_instance" "my_ubuntu_1" {
+   architecture      = (known after apply)
+   arn                = (known after apply)
+   boot_mode         = (known after apply)
+   ena_support        = (known after apply)
+   hypervisor         = (known after apply)
+   id                = (known after apply)
+   image_location     = (known after apply)
+   image_owner_alias  = (known after apply)
+   image_type        = (known after apply)
+   imds_support       = (known after apply)
+   kernel_id          = (known after apply)
+   manage_ebs_snapshots = (known after apply)
+   name              = "MyUbuntu_1"
+   owner_id           = (known after apply)
+   platform           = (known after apply)
+   platform_details   = (known after apply)
+   public             = (known after apply)
+   ramdisk_id         = (known after apply)
+   root_device_name   = (known after apply)
+   root_snapshot_id   = (known after apply)
+   source_instance_id = (known after apply)
+   sriov_net_support  = (known after apply)
+   tags_all           = (known after apply)
+   tpm_support        = (known after apply)
+   uefi_data          = (known after apply)
+   usage_operation    = (known after apply)
+   virtualization_type = (known after apply)

+   ebs_block_device (known after apply)

+   ephemeral_block_device (known after apply)
}
```

Instances (7) [Info](#)

Last updated less than a minute ago [Refresh](#) [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

[All states](#)

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elast
<input type="checkbox"/>	aws-ubuntu	i-0a393c52055d03d07	Running	t2.micro	2/2 checks passed	View alarms	eu-central-1b	ec2-18-185-69-247.eu-...	18.185.69.247	-
<input type="checkbox"/>	my-ubuntu-tem_1	i-0ae89f7f390ff1179	Running	t2.micro	2/2 checks passed	View alarms	eu-central-1a	ec2-18-197-126-218.eu...	18.197.126.218	-
<input type="checkbox"/>	ubuntu	i-0c7236852ba260242	Stopped	t2.micro	-	View alarms	eu-central-1b	-	-	-
<input type="checkbox"/>	av1_ubuntu1	i-0bc6d5039555a5601	Terminated	t2.micro	-	View alarms	eu-central-1b	-	-	-
<input type="checkbox"/>	av1_ubuntu1	i-08b114b4af2bed8d6	Running	t2.micro	2/2 checks passed	View alarms	eu-central-1b	ec2-18-193-254-98.eu-...	18.193.254.98	-
<input type="checkbox"/>	av1_ubuntu1	i-0f6e10588573c812a	Terminated	t2.micro	-	View alarms	eu-central-1b	-	-	-
<input type="checkbox"/>	my-ubuntu-tem_1	i-046a34f3f832754bb	Running	t2.micro	2/2 checks passed	View alarms	eu-central-1c	ec2-35-159-234-79.eu-...	35.159.234.79	-

Select an instance