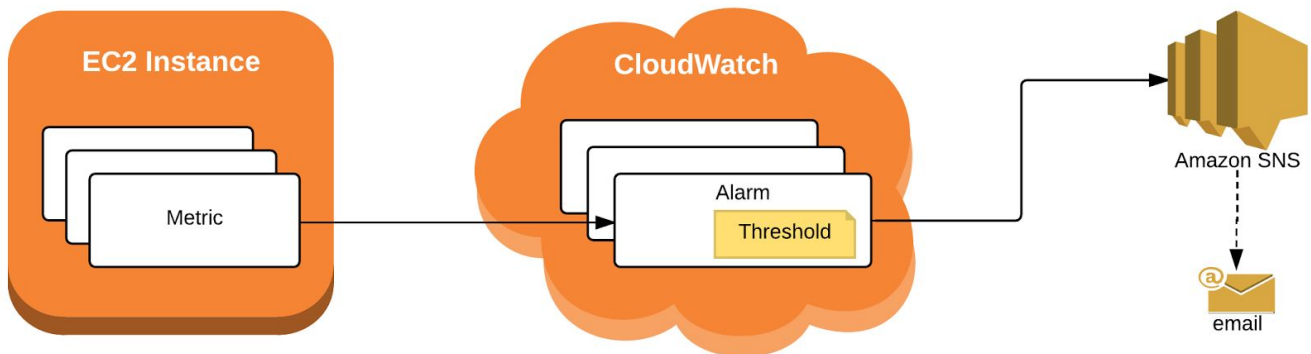


## MODULE 6 – LAB EXERCISES

The lab exercises described below will help you to get practice in both Amazon CloudWatch and Amazon Simple Notification Service in performance monitoring and alarm alerting for your computing instances.

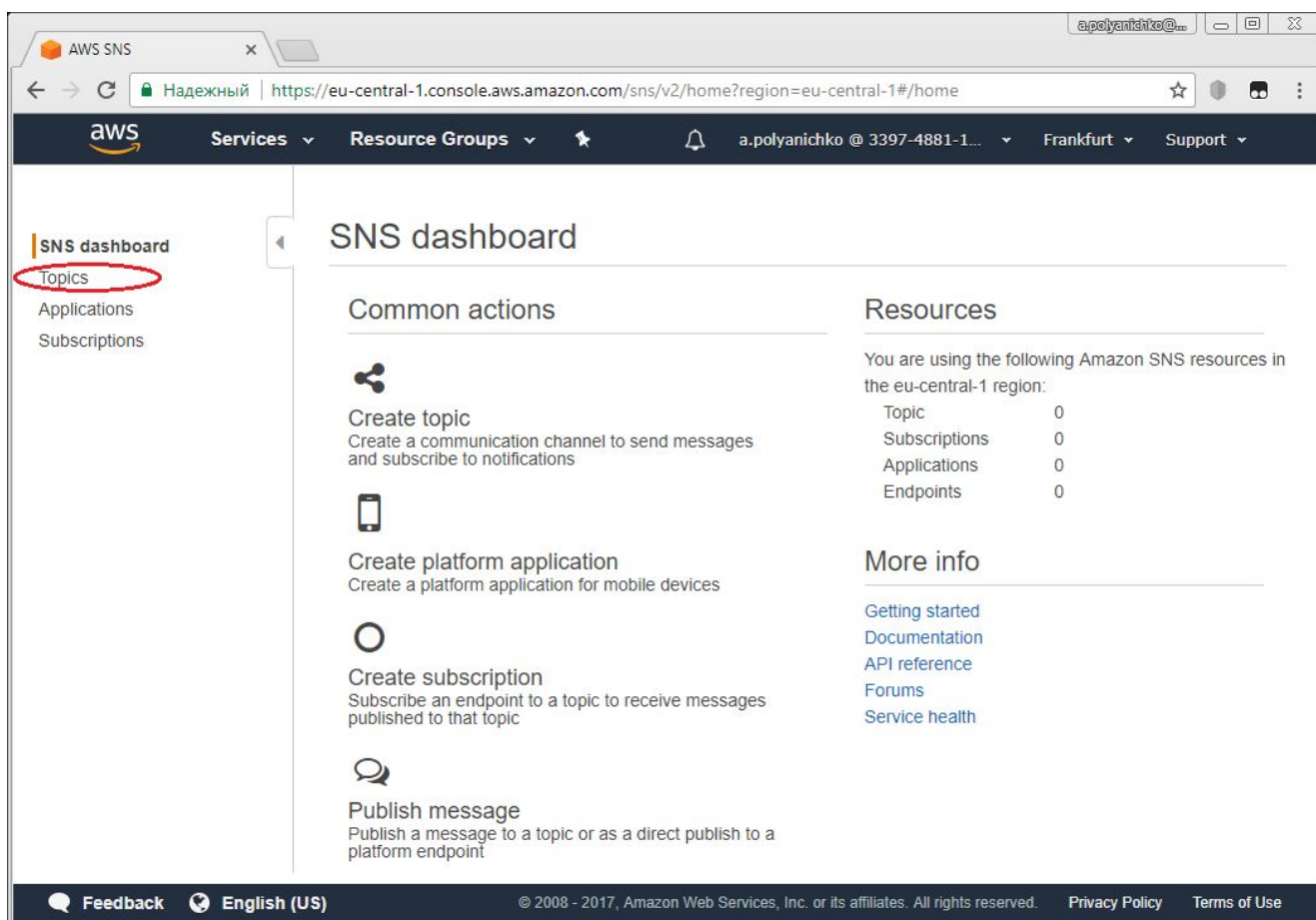
During the hands-on exercises we will use the scheme below as a reference model:



### 1. Create a new SNS Topic

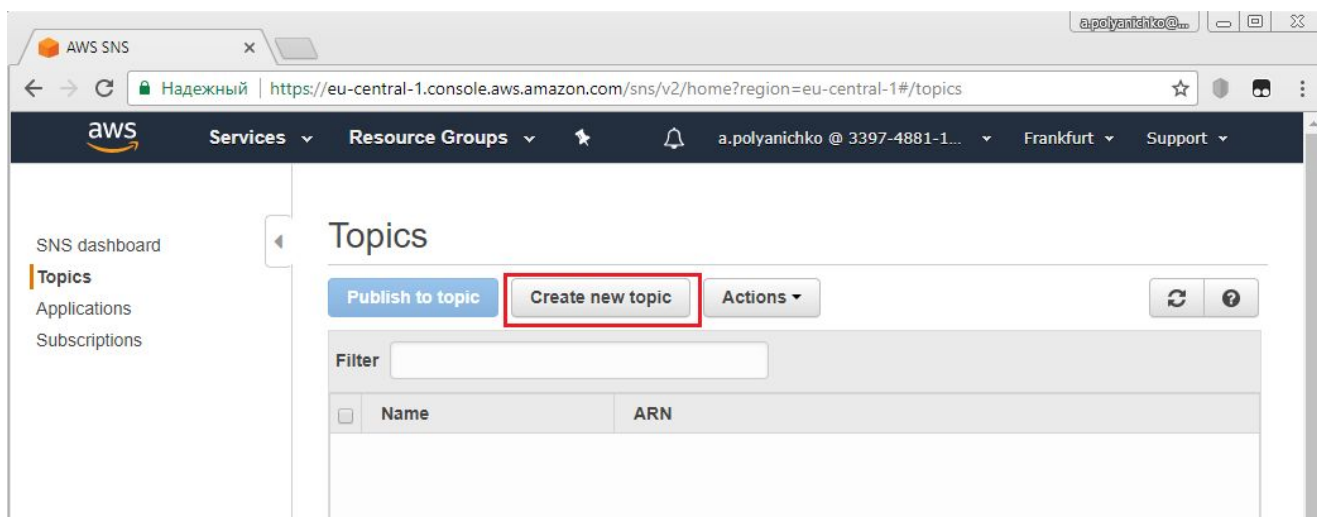
A quite comfortable way for our case is to start our configuration "from the tail" therefore let begin at e-mail notification part.

Please open AWS Management Console and then navigate to SNS main page:



If nothing was created before in Amazon SNS under your account, you will see a welcome page with some brief explanations, just click on “Get started” button.

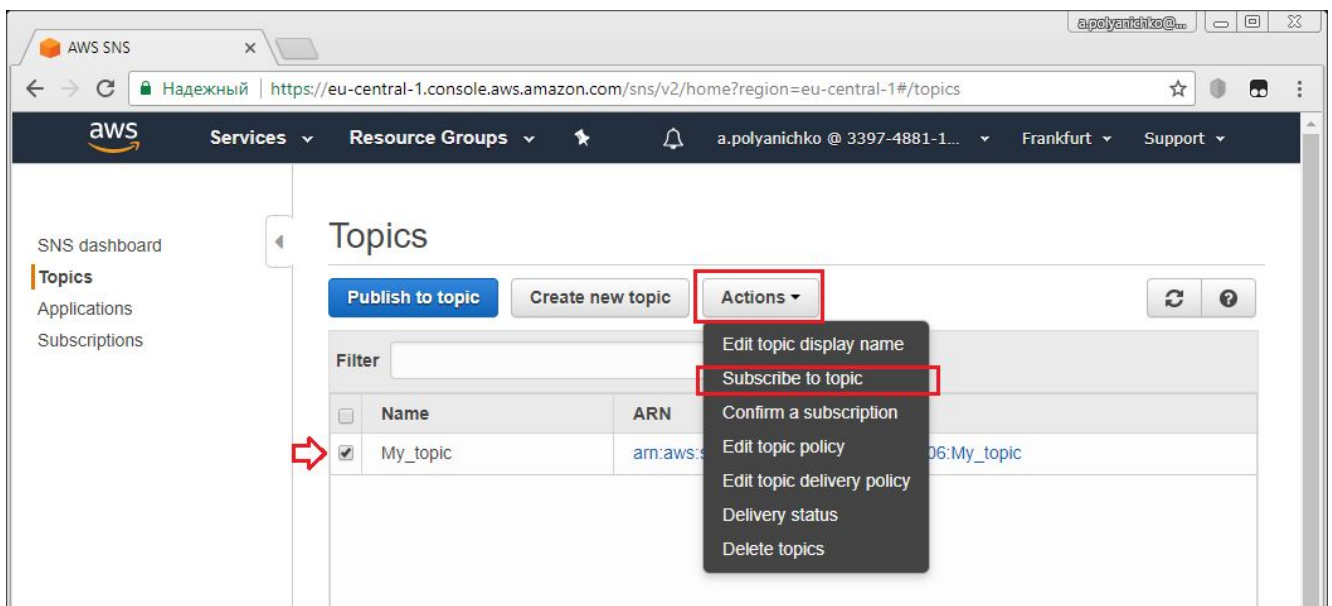
Click on “Topic” item on left-side menu (refer to previous screenshot) to open Topics page and then click on “Create new topic” button:



Specify the name and (optionally) displayed name for new topic, then click on “Create topic” button:

The screenshot shows the 'Create new topic' dialog box. It has a title bar 'Create new topic'. Below the title bar, there is a note: 'A topic name will be used to create a permanent unique identifier called an Amazon Resource Name (ARN)'. There are two input fields: 'Topic name' with the value 'My\_topic' and 'Display name' with the placeholder text 'Enter topic display name. Required for topics with SMS subscriptions.' At the bottom right, there are two buttons: 'Cancel' and 'Create topic', with the 'Create topic' button highlighted by a red rectangle.

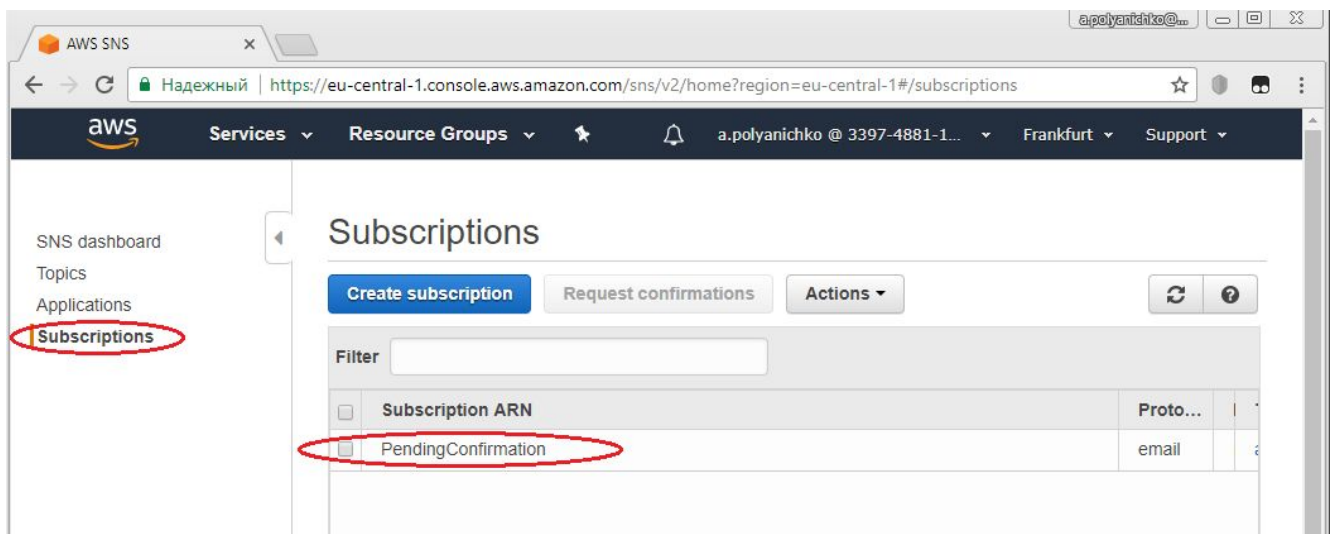
On the next page select your topic in the list then perform “Actions” → “Subscribe to topic” command from Topics top menu:



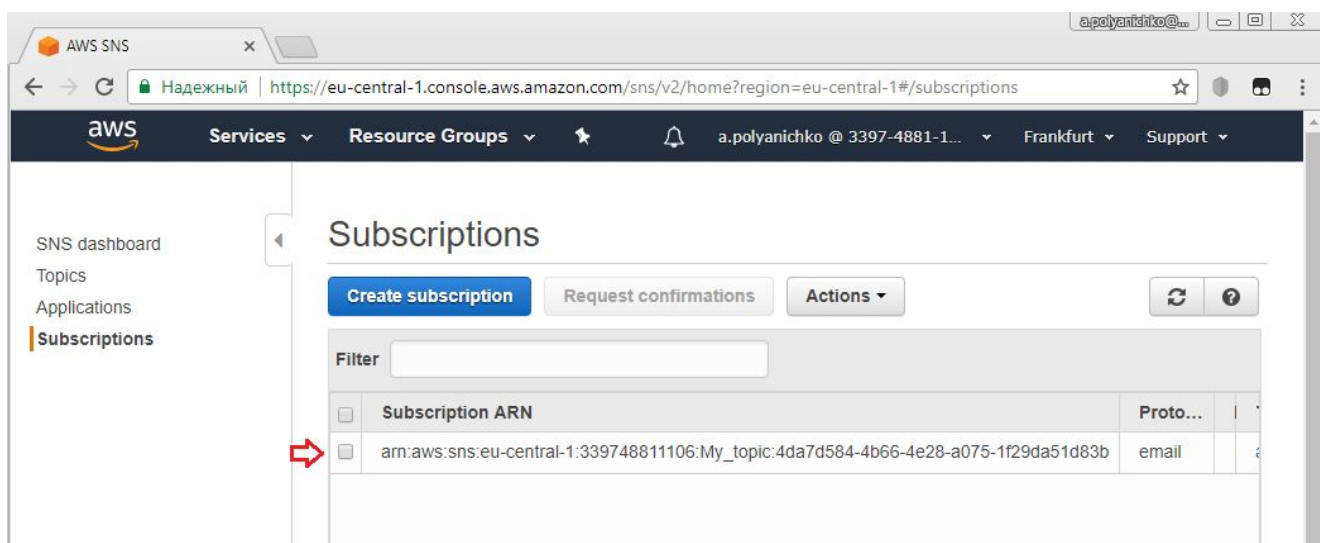
Please select Email protocol and then type the valid e-mail address as endpoint for the subscription:

The screenshot shows the 'Create subscription' dialog box in the AWS SNS console. It has three input fields: 'Topic ARN' with the value 'arn:aws:sns:eu-central-1:339748811106:My\_topic', 'Protocol' set to 'Email' in a dropdown menu, and 'Endpoint' with the value 'rasshua@gmail.com'. At the bottom right, there are two buttons: 'Cancel' and 'Create subscription', with the latter highlighted by a red rectangle.

Click on “Create subscription” button and find your subscription on Subscriptions page as shown below:



At this stage the subscription is waiting for confirmation by e-mail. Please check the mailbox that you have put as endpoint, find the message from AWS Notifications and confirm using of e-mail address. After that your subscription is being active:



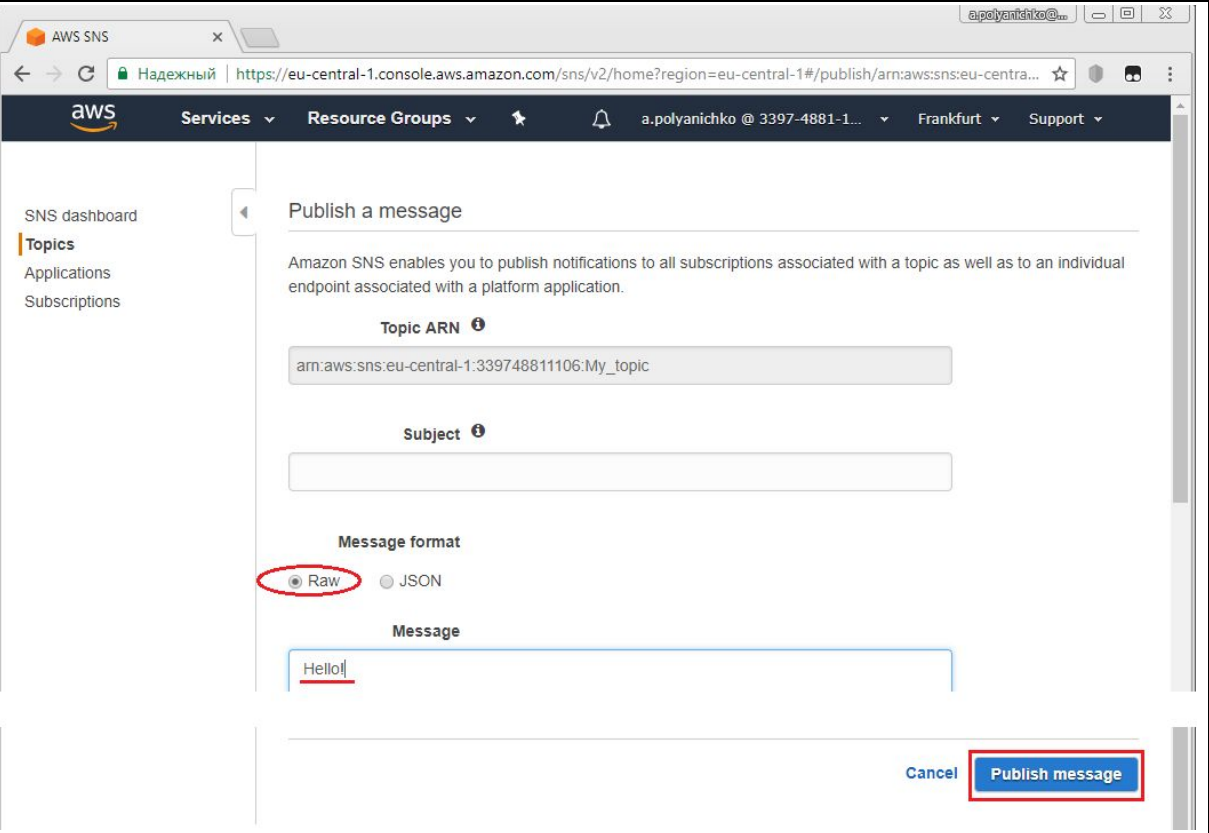
To check that everything is ok with your SNS you may test your topic manually.

Select desired topic in the list as shown below and click on “Publish to topic” button:

The screenshot shows the AWS SNS console in the 'Topics' view. The left navigation pane has 'Topics' selected and circled in red. The main content area shows a table of topics. A red arrow points to the first row of the table.

Name	ARN
My_topic	arn:aws:sns:eu-central-1:339748811106:My_topic

On the next page select “Raw” as message format, type a piece of text in Message textbox and finally click on “Publish message” button at the bottom:



The screenshot shows the AWS SNS console in the 'Publish a message' section. The interface includes a left-hand navigation menu with 'Topics', 'Applications', and 'Subscriptions'. The main content area has a header 'Publish a message' and a description: 'Amazon SNS enables you to publish notifications to all subscriptions associated with a topic as well as to an individual endpoint associated with a platform application.' Below this, there are three input fields: 'Topic ARN' (pre-filled with 'arn:aws:sns:eu-central-1:339748811106:My\_topic'), 'Subject' (empty), and 'Message format' (with 'Raw' selected and 'JSON' as an option). The 'Raw' option is circled in red. Below the 'Message format' section is a 'Message' input field containing the text 'Hello!'. At the bottom right, there are two buttons: 'Cancel' and 'Publish message', with the latter button highlighted by a red rectangle.

SNS dashboard

- Topics
- Applications
- Subscriptions

### Publish a message

Amazon SNS enables you to publish notifications to all subscriptions associated with a topic as well as to an individual endpoint associated with a platform application.

**Topic ARN** ⓘ

arn:aws:sns:eu-central-1:339748811106:My\_topic

**Subject** ⓘ

**Message format**

☒ Raw ☐ JSON

**Message**

Hello!

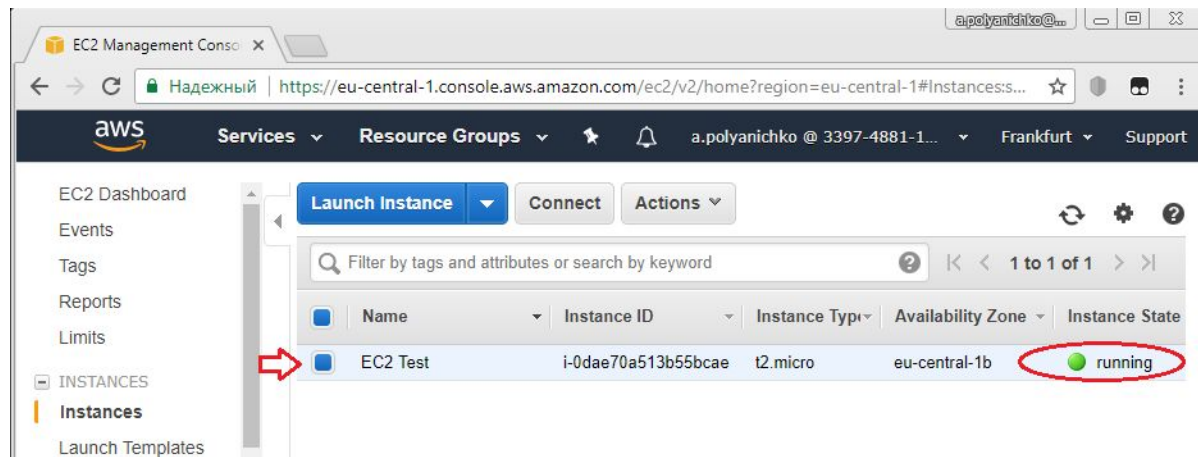
Cancel **Publish message**

You will receive the message on the address that you've specified in the subscription.

## 2. Starting CloudWatch monitoring on the existing EC2 Instance



In the exercise we will use Amazon Linux AMI on EC2 instance which was created in our previous practice. We assume that your EC2 instance exists and is in running state:

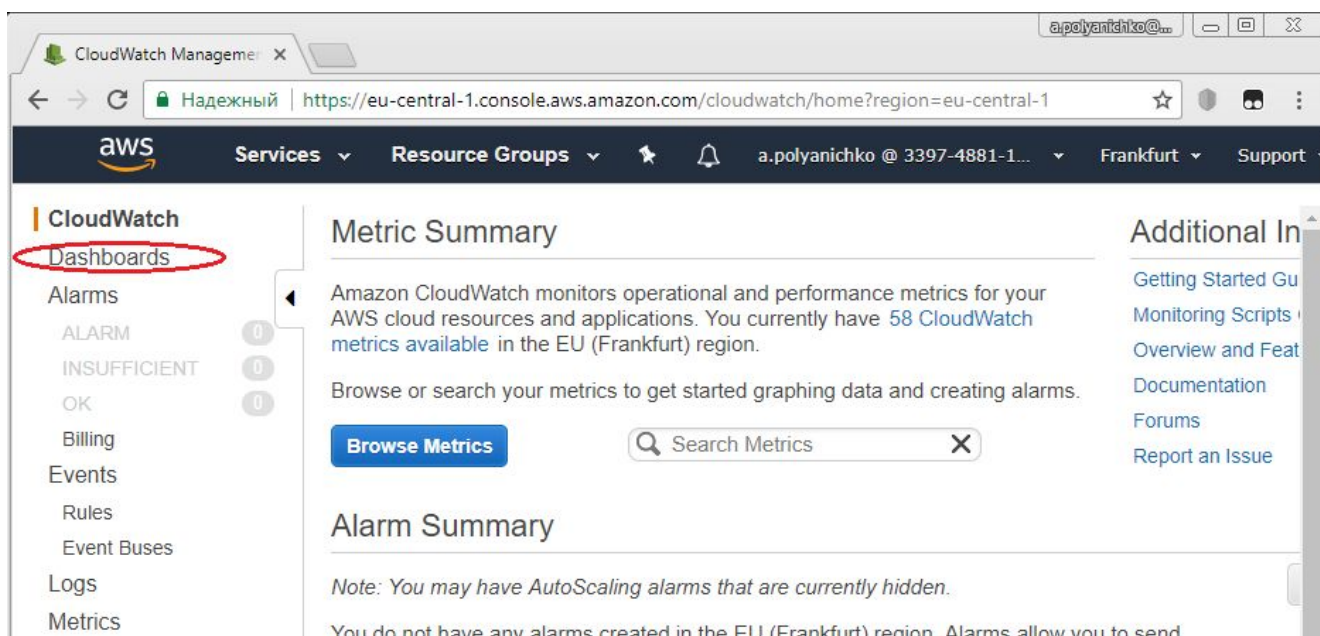


Otherwise please create EC2 instance from scratch as was described in MODULE 3 Lab classwork.

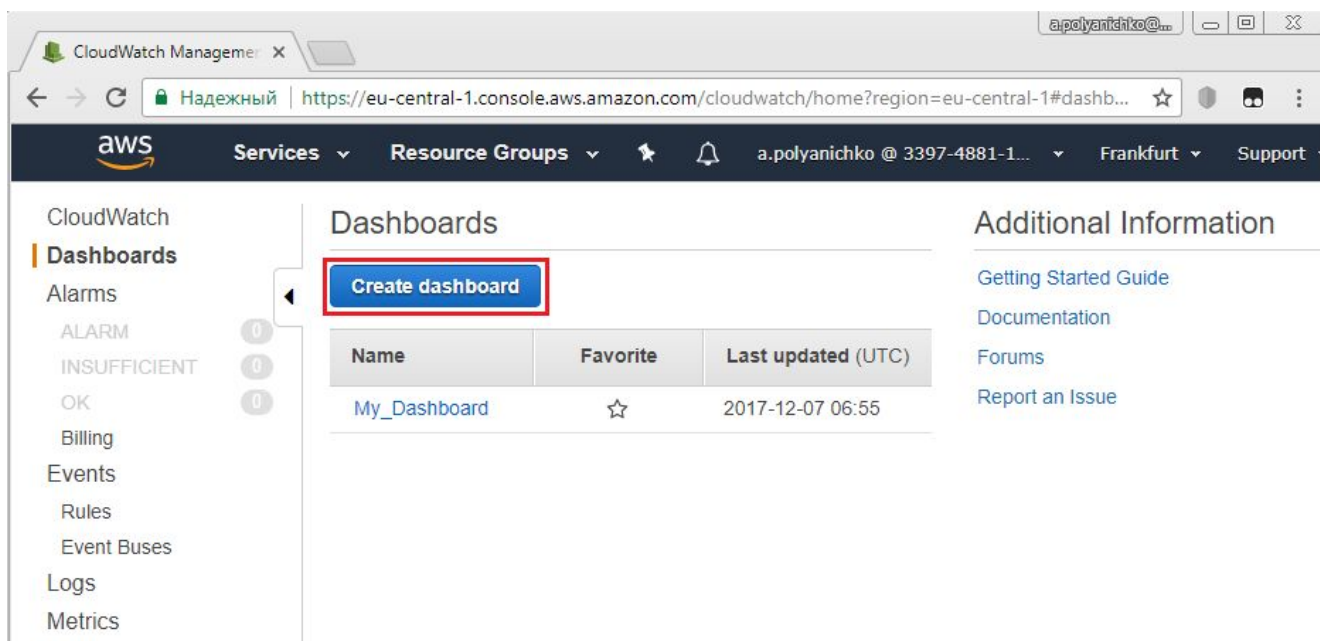
On AWS Management Console please find and click on CloudWatch link:

On CloudWatch main page let start from Dashboard menu item:





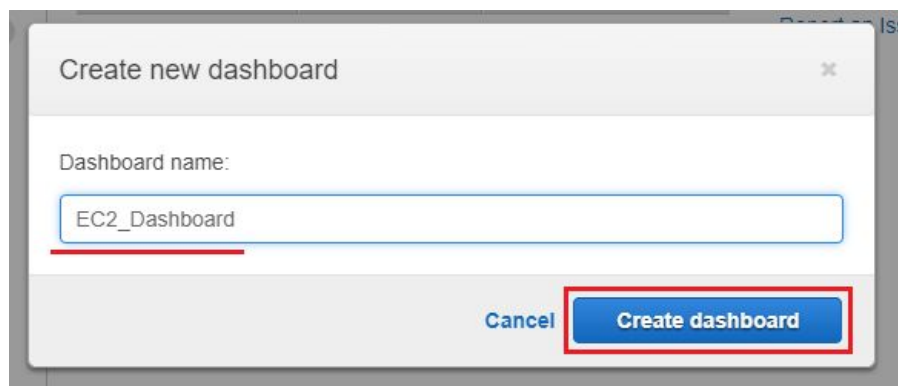
Please click on “Create dashboard” button on the next page:



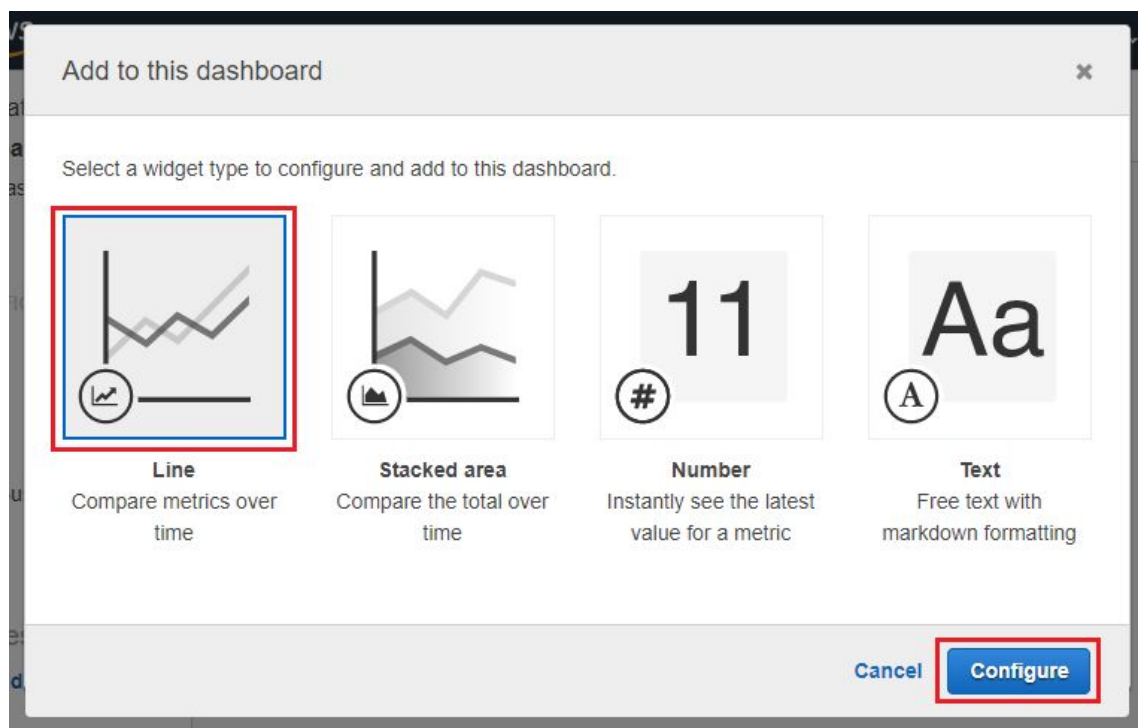
Frankly speaking, the creation of a dashboard is not mandatory for objectives of this exercise. We will pass this step because the dashboard is comfortable tool for metric observation and alarm configuration.

Specify the name for dashboard and then click on “Create dashboard” button below:

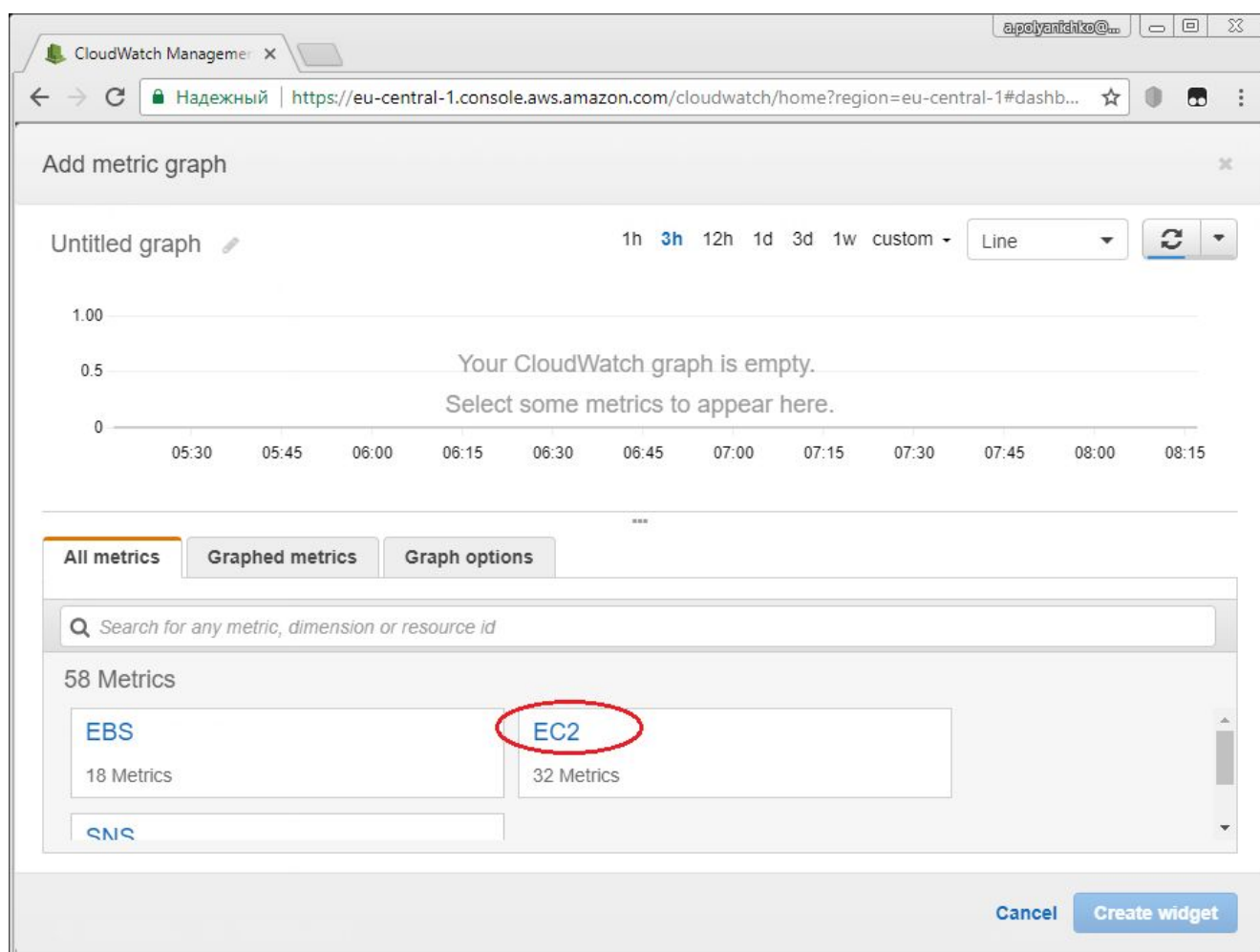




Select a type of widget which will be added on the dashboard and then click “Configure” button:






On the next page you must select the metric (-s) which will be displayed on the widget:



CloudWatch Manager X

Надежный | <https://eu-central-1.console.aws.amazon.com/cloudwatch/home?region=eu-central-1#dashb...>

### Add metric graph

Untitled graph  1h **3h** 12h 1d 3d 1w custom ▾ Line  

1.00  
0.5  
0

Your CloudWatch graph is empty.  
Select some metrics to appear here.

05:30 05:45 06:00 06:15 06:30 06:45 07:00 07:15 07:30 07:45 08:00 08:15

\*\*\*

All metrics Graphed metrics Graph options

Q Search for any metric, dimension or resource id

58 Metrics

EBS 18 Metrics

**EC2** 32 Metrics

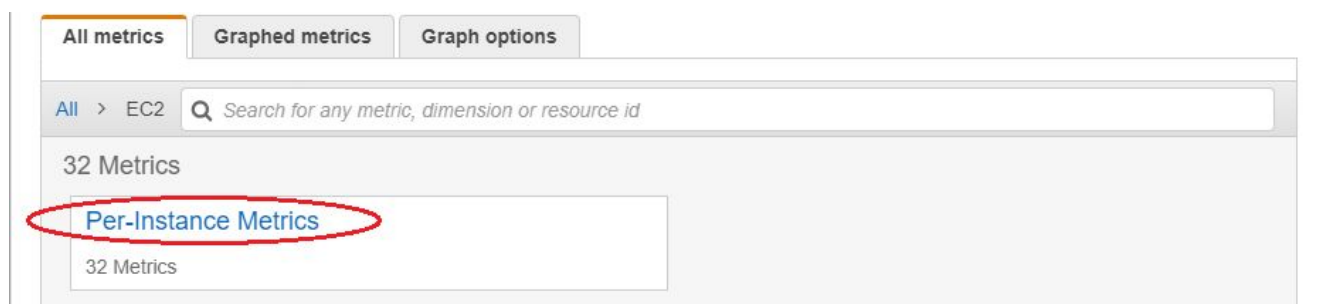
SNS

Cancel Create widget



We are using CPUUtilization metric from our active EC2 for training purpose.

Please click on “EC2” link (shown above) then on “Per-Instance Metric” link:



All metrics Graphed metrics Graph options

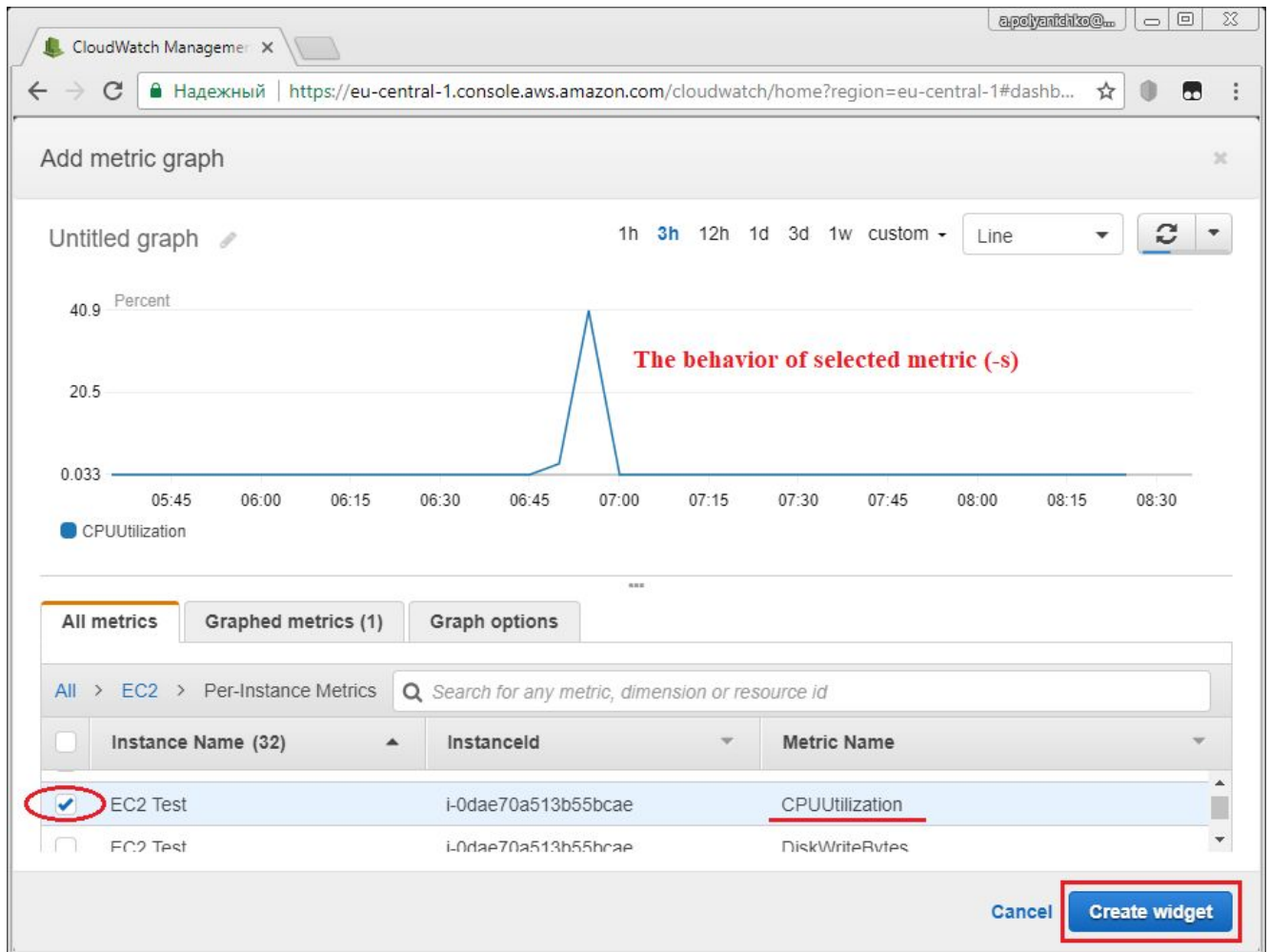
All > EC2 Q Search for any metric, dimension or resource id

32 Metrics

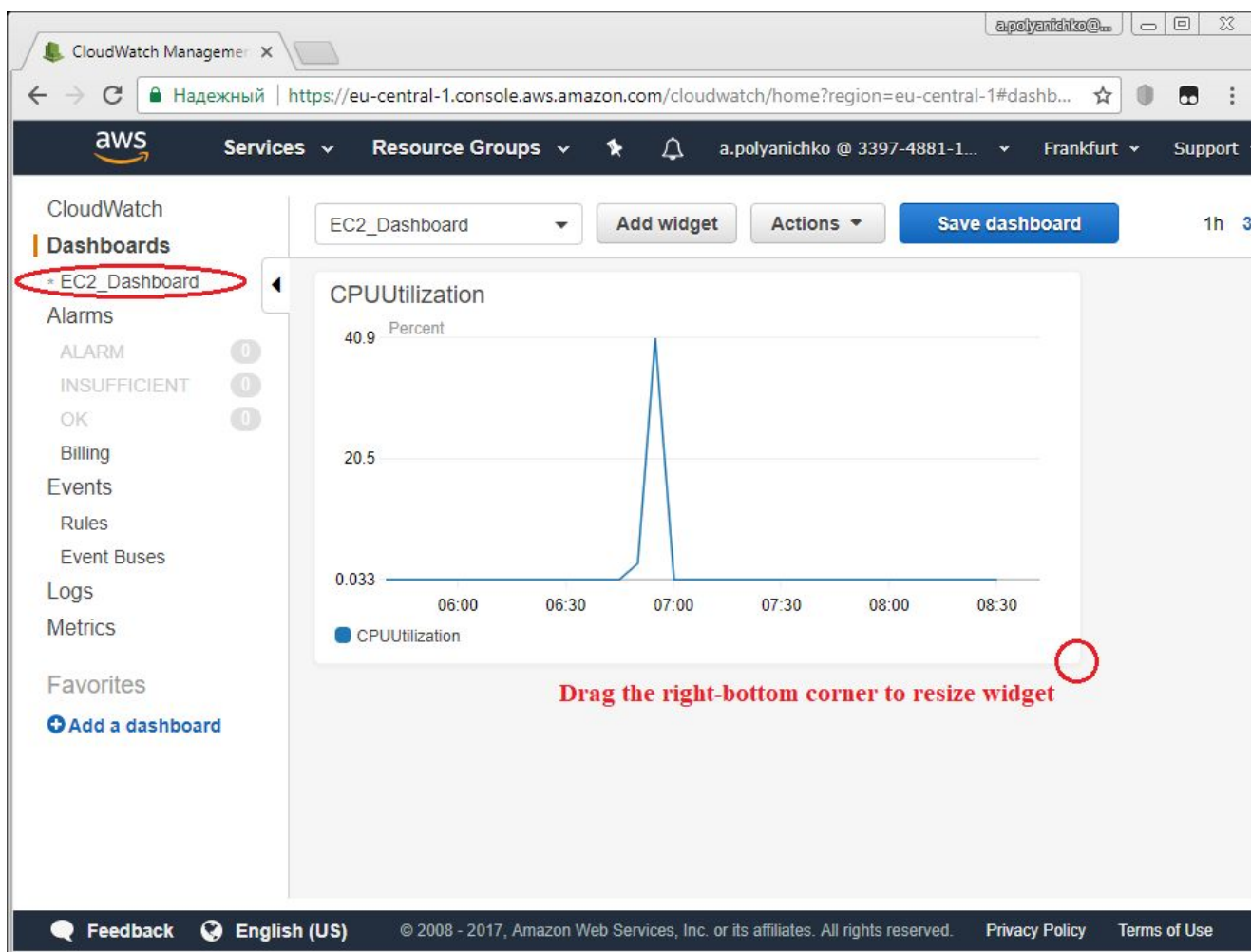
**Per-Instance Metrics**

32 Metrics

and find and select CPUUtilization metric for your desired EC2:



Finally click on “Create widget” button as shown above and ensure that your dashboard displays desired metric:



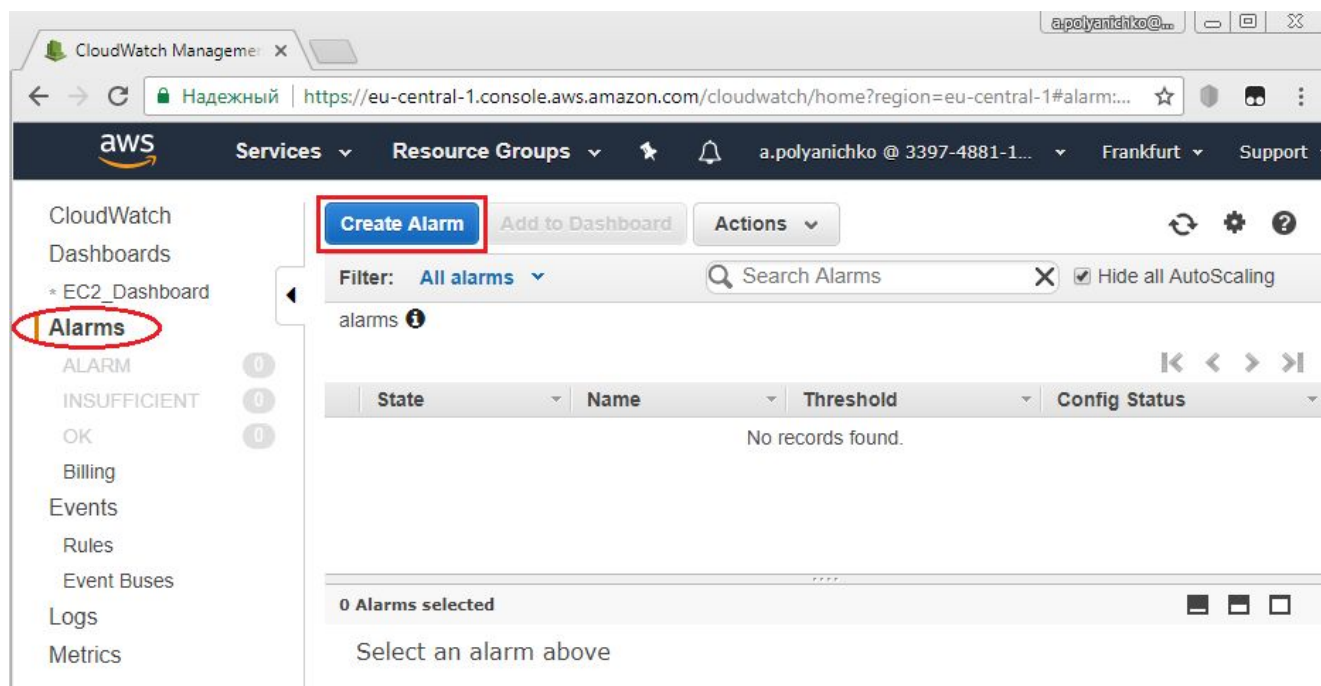
Please take in account that the time on X-Axis is displayed in UTC format by default.

So your dashboard is ready for use and now you may start to create alarm in CloudWatch.

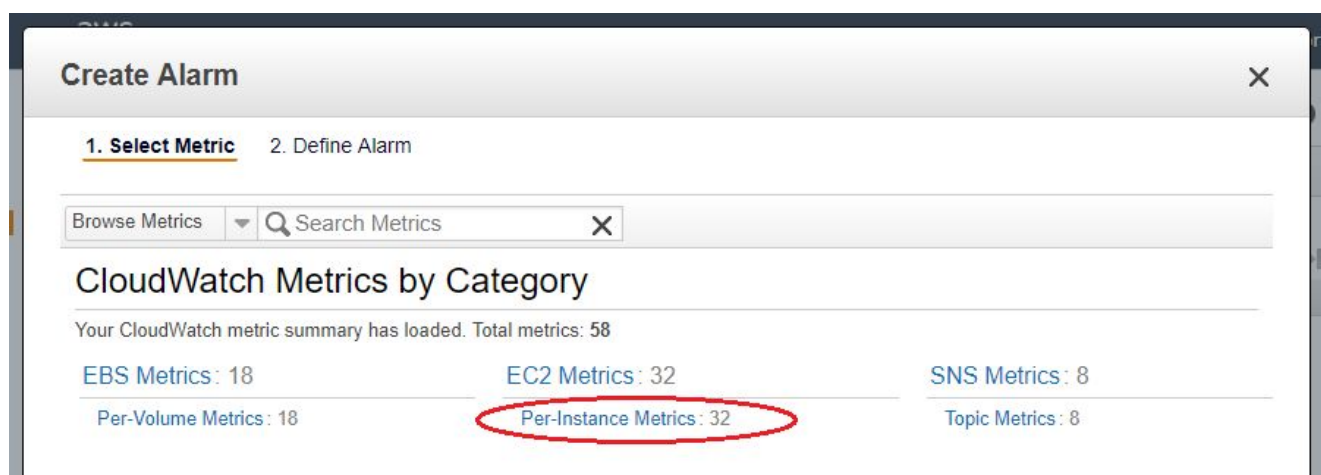


At this point please note average maximal value for CPU utilization using your dashboard and widget. CPU utilization in “idle” mode will be necessary when you will configure alarm threshold. In our testing lazy CPU utilization values were less than 5%.

Switch to “Alarm” item on the left-side menu and click on “Create alarm” button:



On the next page select desired metric group:



then select desired metric (1), define measurement interval (2) and optionally minimal limit for Y-axis (3), and finally click on “Next” button (4):

**Create Alarm**

1. Select Metric 2. Define Alarm

EC2 Search Metrics X 1 to 32 of 32 metrics

1 ☐ i-0dae70a513b55bcae EC2 Test CPUSurplusCreditsCharged

☒ i-0dae70a513b55bcae EC2 Test CPUUtilization

☐ i-0dae70a513b55bcae EC2 Test DiskReadBytes

Title: CPUUtilization Average 1 Minute 2

40.9 Percent

20.5

0

00:00 03:00 06:00 09:00

i-0dae70a513b55bcae (EC2 Test)

Update Graph

Time Range

Relative Absolute UTC (GMT)

From: 12.01 hours ago

To: 0 hours ago

Zoom: 1h | 3h | 6h | 12h | 1d | 3d | 1w | 2w

Left Y-axis

Limits Min 0 Max

3 0 Auto

Cancel Previous Next Create Alarm

On the next step you must specify alarm threshold (name and value) as shown in example below:

Create Alarm

1. Select Metric

2. Define Alarm

### Alarm Threshold



Provide the details and threshold for your alarm. Use the graph on the right to help set the appropriate threshold.

**Name:**

**Description:**


**Whenever:** CPUUtilization

**is:**

**for:** 1  out of  datapoints 

### Additional settings

Provide additional configuration for your alarm.

**Treat missing data as:**  



And finally you specify consecutive actions for the alarm state as shown below:

**Create Alarm** [X]

1. Select Metric    **2. Define Alarm**

### Actions

Define what actions are taken when your alarm changes state.

Notification [Delete]

Whenever this alarm: State is ALARM

Send notification to: My\_topic [New list] [Enter list] [i]

Email list: pol.andre456@gmail.com

Notification [Delete]

Whenever this alarm: State is OK

Send notification to: My\_topic [New list] [Enter list] [i]

Email list: pol.andre456@gmail.com

[+ Notification]   [+ AutoScaling Action]   [+ EC2 Action]

[Cancel]   [Previous]   [Next]   **Create Alarm**

Feedback   English (US)   © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.   Privacy Policy   Terms of Use



You must set the name of your SNS topic which was created before as the target to send notifications to.



Use “+Notification” button to add one more notification to the list (by default there is only one string there).

Finally click on “Create alarm” button (shown above) to create and activate alarm and watch Alarm section:

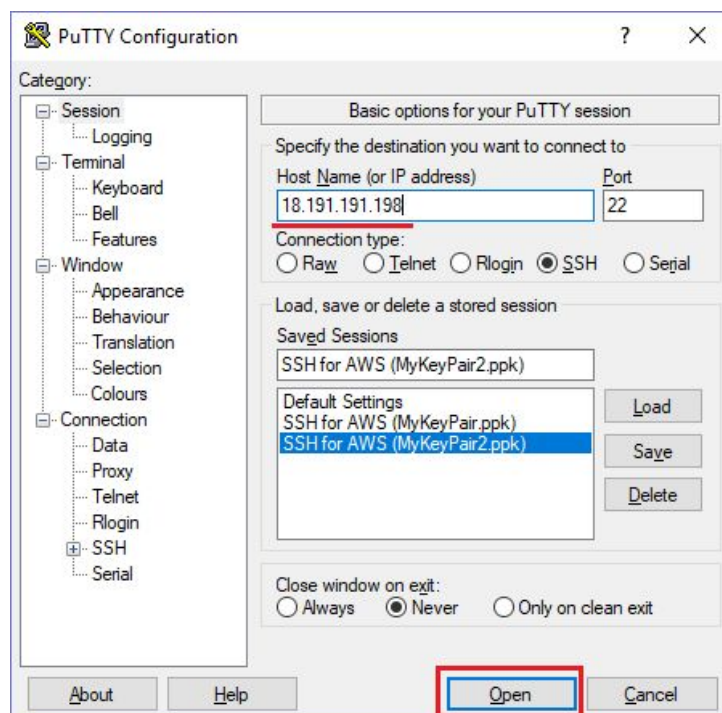
If everything is ok with your SNS topic, you will receive “OK” notification on e-mail address which was specified in SNS subscription.



“INSUFFICIENT” value may be displayed for your alarm as intermediate state in short time just after creation. It will not be notified on e-mail.

Now let increase the computing load for our EC2 by calculating the number "pi" with an accuracy greater than 1000 decimal places.

Connect to your EC2 instance from PuTTY via SSH using public IP address as was described in MODULE 3 hands-on exercises:




R1

```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Thu Dec  7 06:56:21 2017 from 31.13.22.89
```

```
 _ | _ | )
 _ | ( _ /  Amazon Linux AMI
 _|\_|_|_|
```

```
https://aws.amazon.com/amazon-linux-ami/2017.09-release-notes/
No packages needed for security; 1 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-43-55 ~]$
[ec2-user@ip-172-31-43-55 ~]$
[ec2-user@ip-172-31-43-55 ~]$ █
```

and run the following command sequence:

	<pre>[ec2-user@ip-172-31-43-55 ~]\$ bc -lq scale=1000 4*a(1)</pre>
---	--

The output example is shown below:

```
[ec2-user@ip-172-31-43-55 ~]$
[ec2-user@ip-172-31-43-55 ~]$
[ec2-user@ip-172-31-43-55 ~]$ bc -lq
scale=1000
4*a(1)
3.141592653589793238462643383279502884197169399375105820974944592307\
81640628620899862803482534211706798214808651328230664709384460955058\
22317253594081284811174502841027019385211055596446229489549303819644\
28810975665933446128475648233786783165271201909145648566923460348610\
45432664821339360726024914127372458700660631558817488152092096282925\
40917153643678925903600113305305488204665213841469519415116094330572\
70365759591953092186117381932611793105118548074462379962749567351885\
75272489122793818301194912983367336244065664308602139494639522473719\
07021798609437027705392171762931767523846748184676694051320005681271\
45263560827785771342757789609173637178721468440901224953430146549585\
37105079227968925892354201995611212902196086403441815981362977477130\
99605187072113499999983729780499510597317328160963185950244594553469\
08302642522308253344685035261931188171010003137838752886587533208381\
42061717766914730359825349042875546873115956286388235378759375195778\
18577805321712268066130019278766111959092164201988

quit
[ec2-user@ip-172-31-43-55 ~]$
[ec2-user@ip-172-31-43-55 ~]$
```

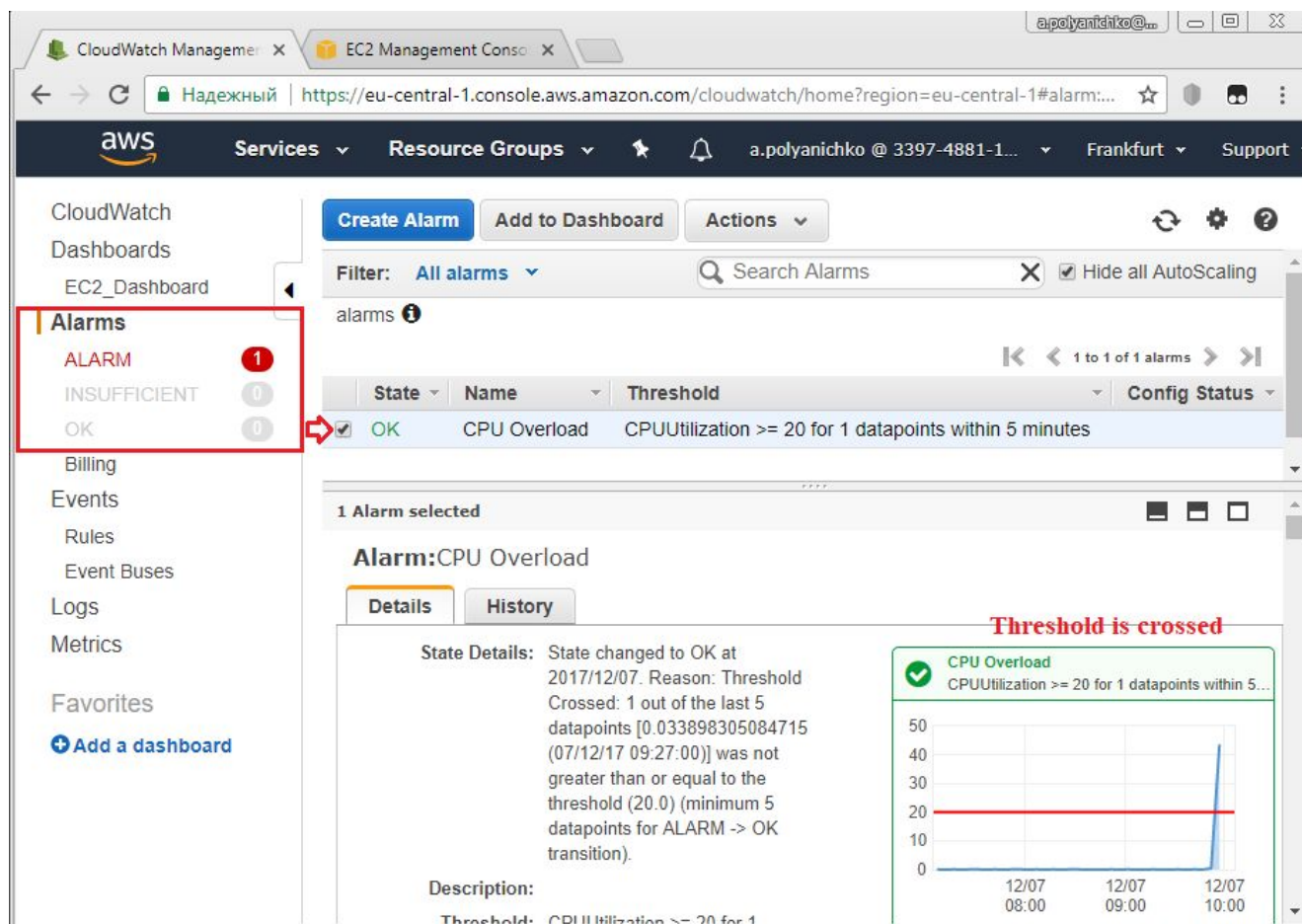
Repeat the sequence of commands, replacing the value of Scale by 10000:

```
[ec2-user@ip-172-31-43-55 ~]$
[ec2-user@ip-172-31-43-55 ~]$ bc -lq
scale=10000
4*a(1)
```



The computing will take longer time comparing to previous example. You may interrupt the process by <Ctrl>+<C> combination.

After a few minutes you will receive ALARM notification on e-mail address and the status of alarm will be changed similar to example below:



You may examine your dashboard to see exactly values of CPU utilization in the time



If you will be doing nothing during a few minutes and, of course, computing in EC2 was completed or interrupted, shortly you will receive OK notification on e-mail address and the status of alarm will be changed back to “green” state:

The screenshot displays the AWS CloudWatch Alarms console in the eu-central-1 region. The left-hand navigation menu is visible, with the 'Alarms' section highlighted by a red box. Within this menu, the 'OK' status is indicated by a green circle with the number '1'. A red arrow points from this status indicator to the main content area. The main area shows a table of alarms with the following details:

State	Name	Threshold
OK	CPU Overload	CPUUtilization >= 20 for 1 datapoints within 5 minutes

Below the table, the 'Details' tab for the 'Alarm: CPU Overload' is selected. The 'State Details' section reports: 'State changed to OK at 2017/12/07. Reason: Threshold Crossed: 1 out of the last 5 datapoints [0.033333333333333303 (07/12/17 10:02:00)] was not greater than or equal to the threshold (20.0) (minimum 5 datapoints for ALARM -> OK transition)'. A description at the bottom states: 'Threshold: CPUUtilization >= 20 for 1'. To the right, a line graph titled 'CPU Overload' shows CPU utilization over time, with a red horizontal threshold line at 20. A blue line shows a single data point spike that has fallen below the threshold. A red text annotation above the graph reads: 'The metric is under threshold again:'. The footer of the console shows the date '© 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.' and links for 'Feedback', 'English (US)', 'Privacy Policy', and 'Terms of Use'.