

# PROJEKTOWANIE SYSTEMÓW OBIEKTOWYCH I ROZPROSZONYCH

## LABORATORIUM 2

### *Amazon Elastic Cloud Compute (EC2)*

wersja 1.2

przygotował:  
dr inż. Radosław Adamus

## Cel:

Celem laboratorium jest zapoznanie z:

1. Mechanizmami definiowania instancji EC2 na podstawie obrazu oraz zarządzania nią z poziomu konsoli webowej.
2. Wykorzystywaniem instancji EC2 jako miejsca uruchamiania aplikacji
3. Mechanizmami programowymi zarządzania instancją EC2.

## Wymagania wstępne:

1. Posiadanie konta na platformie Github.
2. Skonfigurowane konto AWS
3. Zapoznanie się z Tutorial - Amazon Web Services EC2 (<http://www.youtube.com/watch?v=YtwYcpZeoXA> )
4. Zapoznanie się z platformą nodeJS (<http://stackoverflow.com/questions/2353818/how-do-i-get-started-with-node-js>)
5. Zapoznanie się z opisem przykładowej aplikacji i sposobem jej rozbudowy (<https://github.com/amgnet-weeia/awslab>)

## Narzędzia:

1. Przeglądarka internetowa
2. Putty, PuttyGen
3. Git

## Reguły wykonywania ćwiczeń laboratoryjnych:

1. Na zakończenie laboratorium należy wyłączyć wszystkie uruchomione instancje EC2.
2. Otrzymane programowe klucze dostępowe (szczególnie Secret Access Key) nie mogą zostać upublicznione. Jeżeli zaistnieje takie podejrzenie należy natychmiast poinformować o tym fakcie prowadzącego.

## Wstęp:

Instancje EC2 to podstawowe jednostki obliczeniowe w chmurze Amazon. Reprezentują one logiczny komputer z zainstalowanym systemem operacyjnym oraz posiadającym określone zasoby sprzętowe oraz sieciowe. Właściciel instancji, użytkownik AWS, może samodzielnie nią zarządzać oraz ponosi koszty jej działania. Koszty zależne są przede wszystkim od typu instancji określającego jej "pojemność obliczeniową" (moc obliczeniowa, ilość pamięci, dostępna przepustowość sieci) oraz od czasu jej działania. Oprócz tego możliwe są dodatkowe koszty wynikające z opłat licencyjnych uruchomionego oprogramowania systemowego oraz aplikacji.

# Opis laboratorium:

## 1. Uruchomienie aplikacji na instancji EC2

Pierwszym zadaniem w ramach laboratorium będzie uruchomienie przykładowej aplikacji na uruchomionej uprzednio instancji EC2. Przykładowa aplikacja wykorzystuje środowisko nodeJS (<http://nodejs.org/>) oraz przechowywana jest na platformie GitHub (<https://github.com/>). Z tego powodu instancja EC2 powinna być uruchomiona na podstawie odpowiednio skonfigurowanego obrazu systemu (AMI).

### 1.1 Konfiguracja i uruchamianie instancji EC2

Przykładowy (i bezpłatny) obraz posiadające preinstalowane komponenty, spełniające wymagania przykładowej aplikacji, dostępny jest w ramach usługi Amazon Marketplace: ([https://aws.amazon.com/marketplace/pp/B00GXYEEA4/ref=srh\\_res\\_product\\_title?ie=UTF8&sr=0-2&qid=1393245935872](https://aws.amazon.com/marketplace/pp/B00GXYEEA4/ref=srh_res_product_title?ie=UTF8&sr=0-2&qid=1393245935872)).

Na stronie obrazu wciśnij przycisk *Continue*. Następnie wybierz zakładkę uruchamiania ręcznego (Manual Launch) pozwalającą na uruchamianie instancji z wykorzystaniem konsoli EC2. Wybierz tę opcję dla regionu US-West (Oregon). Podczas konfiguracji instancji zadбай o następujące ustawienia (pozostałe pozostaw domyślne):

- Tag Instance: Key - Name - w polu *Value* wpisz swój login.
- Configure Security Group: wśród istniejących "security group" znajdź grupę opisaną "default VPC security group".

Przy pierwszym uruchomieniu będziesz musiał wygenerować nową parę kluczy ssh. Nazwij je swoim loginem.

### 1.2 Zadania

1. Sklonuj przykładową aplikację <https://github.com/amgnet-weeia/awslab> na swoje konto GitHub (operacja Fork w interfejsie platformy).

2. Uruchom instancję EC2 a podstawie opisu z punktu 1.1..

3. Zaloguj się na instancję

(<http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstances.html>)

4. Po zalogowaniu pobierz przykładową aplikację ze swojego konta GitHub (polecenie w konsoli *git clone [adres repozytorium]*).

5. Zainstaluj zależności wymagane do uruchomienia aplikacji (polecenie w konsoli *npm install* w folderze aplikacji).

6. Uruchom aplikację poleceniem *node app.js* i przetestuj jej działanie za pośrednictwem przeglądarki.

## 2. Programowa komunikacja z instancją EC2

W tej części laboratorium będziesz modyfikował aplikację. Do tego celu najlepiej będzie sklonować swoje repozytorium na lokalny komputer. Po wprowadzeniu zmian należy je zatwierdzić (*git commit*) do repozytorium i wypchnąć (*git push*) zmiany do repozytorium zdalnego (na platformie GitHub). Zmiany te będzie można następnie pobrać na instancję EC2 (*git pull*).

Dodaj do aplikacji funkcjonalność pozwalającą na pobieranie i wyświetlanie informacji nt. uruchomionej instancji EC2. Do tego potrzebne będzie wykorzystanie modułu AWS SDK (<http://aws.amazon.com/sdkfornodejs/>). .

1. Instalacja za pomocą aplikacji Node Package Manager (npm) z poziomu konsoli w folderze projektu:

```
> npm install aws-sdk --save
```

Aplikacja wymaga indywidualnej konfiguracji uwierzytelnień AWS. W tym celu należy, w folderze projektu, utworzyć plik `config.json` i wprowadzić poniższą zawartość (podstawiając otrzymane klucze dostępowe miejsce `akid` oraz `secret`).

```
{  "accessKeyId":  "akid",  "secretAccessKey":  "secret",  "region":  "us-west-2" }
```

---

**UWAGA! Ponieważ wykorzystujemy publiczne konta na platformie GitHub, informacje zawarte w pliku `config.json` NIE MOGĄ znaleźć się w repozytorium systemu Git. Spowodowałoby to upublicznienie danych uwierzytelniających i co za tym idzie narażenie konta na wykorzystanie przez osoby nieuprawnione. Przykładowy projekt który wykorzystujesz jest tak skonfigurowany, aby plik o nazwie `config.json` był ignorowany przez mechanizm śledzenia zmian (zapoznaj się z zawartością pliku `.gitignore`). Wystarczy jednak literówka, aby to zabezpieczenie nie zadziałało.**

---

Modyfikacja aplikacji powinna odbywać się zgodnie ze sposobem opisanym w pliku `README.md`. W pliku źródłowym reprezentującym dodawany skrypt należy ponadto dodać następujące instrukcje, inicjujące wykorzystanie funkcjonalności modułu AWS SDK:

```
var AWS = require('aws-sdk');

AWS.config.loadFromPath('./config.json');

//... function(
```

Do pobierania informacji na temat instancji EC2 należy wykorzystać następującą funkcjonalność <http://docs.aws.amazon.com/AWSJavaScriptSDK/latest/AWS/EC2.html#describeInstances-property>

### **3. Programowe tworzenie instancji EC2**

Dodaj do aplikacji funkcjonalność (np. w postaci przycisku) pozwalającej na uruchomienie instancji EC2 na podstawie obrazu AMI:

<http://docs.aws.amazon.com/AWSJavaScriptSDK/latest/AWS/EC2.html#runInstances-property> i wyświetlenie, w rezultacie działania, jej adresu IP/DNS.