Plan 28 dni i laboratorium AWS w Zielonej

Poziom: podstawy

Dotyczy: egzamin SAA-C03, AWS Organizations, IAM Identity Center, AWS Budgets, AWS CLI v2, AWS CloudShell, AWS CloudFormation, region eu-central-1

Czas: ~120 min na postawienie laboratorium

Plan dzień po dniu na cztery tygodnie, od pierwszego logowania do egzaminu, i laboratorium, które stawiamy jednym poleceniem i usuwamy drugim. Wracamy tu codziennie, żeby sprawdzić, co dziś, i raz na początku, żeby założyć konta, budżet i sieć.

ZAŁOŻENIA SCENARIUSZA

w tym ćwiczeniu laboratorium AWS jest osobną organizacją AWS, niepowiązaną z kontem archiwum ze strony Zimne archiwum w chmurze: Azure Blob Storage i Amazon S3 Glacier i bez połączenia z siecią Zielonej. Region to `eu-central-1` (Frankfurt), sieć laboratorium to VPC 10.60.0.0/16 w trzech strefach, adres nie nakłada się na 10.20.0.0/16 w serwerowni ani na 10.30.0.0/24 w Azure. Połączenie VPN z `fw01` pojawia się dopiero w ćwiczeniu projektowym sklepu B2B jako wariant na papierze.

Skróty użyte na tej stronie (18)

Skrót	Co znaczy
AWS	Amazon Web Services: chmura Amazonu
B2B	business to business: sprzedaż i systemy skierowane do innych firm, nie do klientów indywidualnych
CLI	Command Line Interface: praca przez wpisywanie komend w terminalu
DNS	Domain Name System: książka telefoniczna internetu, zamienia nazwy na adresy IP
ESL	English as a Second Language: dostosowanie egzaminu AWS dla osób, dla których angielski nie jest językiem ojczystym, daje 30 minut więcej
GB	Gigabyte: miliard bajtów, jednostka pojemności pamięci i plików, tu najczęściej rozmiar modelu albo pamięci karty graficznej

Skrót	Co znaczy
IAM	Identity and Access Management: zarządzanie tożsamościami i uprawnieniami (w AWS: użytkownicy, role, polityki)
IPv4	Internet Protocol wersja 4: adresy zapisane czterema liczbami z kropkami, na przykład 10.20.10.77
MFA	Multi-Factor Authentication: logowanie drugim składnikiem, np. kodem z telefonu
NAT	Network Address Translation: podmiana adresów IP na granicy sieci
SAA-C03	kod wersji egzaminu AWS Solutions Architect Associate zdawanej od 30 sierpnia 2022, aktualnej w październiku 2026
SAML	Security Assertion Markup Language: standard logowania jednokrotnego, w którym dostawca tożsamości potwierdza użytkownika aplikacji
SCP	Service Connection Point: obiekt w Active Directory, po którym komputery domenowe znajdują swój tenant Entra ID przy dołączaniu hybrydowym
SSO	Single Sign-On: jedno logowanie do wielu systemów
USD	dolar amerykański, waluta cenników AWS
VPC	Virtual Private Cloud: prywatna sieć wirtualna klienta w chmurze AWS, w której stawiamy maszyny i zakładamy tablice tras
VPN	Virtual Private Network: szyfrowany tunel do sieci firmowej
YAML	YAML Ain't Markup Language: format plików konfiguracyjnych oparty na wcięciach

Fiszka

Komendy `aws` wykonujemy w AWS CloudShell albo w powłoce bash z AWS CLI v2 po zalogowaniu przez IAM Identity Center. Nazwy `IdKonta`, `IdKorzenia`, `NazwaProfilu`, `AdresEmail` i podobne zastępujemy własnymi. Fiszki idą w kolejności stawiania laboratorium.

TYLKO ODCZYT

Sprawdzenie planu konta Free Tier: aws freetier get-account-plan-state

```
aws freetier get-account-plan-state
```

Pokazuje, czy konto jest na planie darmowym, czy płatnym, i ile zostało kredytów. Konto na planie darmowym nie ma dostępu do części usług, a dołączenie do AWS Organizations samo przełącza je na plan płatny.

AWS = Amazon Web Services: chmura Amazonu · **aws** = AWS CLI: narzędzie chmury Amazonu

ZMIENIA SYSTEM

Utworzenie organizacji ze wszystkimi funkcjami: aws organizations create-organization

```
aws organizations create-organization --feature-set ALL
```

Wykonujemy na koncie, które ma zostać kontem zarządzającym. Tryb `ALL` jest potrzebny do polityk SCP i do IAM Identity Center dla całej organizacji.

SCP = Service Connection Point: obiekt w Active Directory, po którym komputery domenowe znajdują swój tenant Entra ID przy dołączaniu hybrydowym · **IAM** = Identity and Access Management: zarządzanie tożsamościami i uprawnieniami (w AWS: użytkownicy, role, polityki) · **aws** = AWS CLI: narzędzie chmury Amazonu

TYLKO ODCZYT

Odczyt identyfikatora korzenia organizacji: aws organizations list-roots

```
aws organizations list-roots --query "Roots[0].Id" --output text
```

Przykład (kształt wyniku, wartości z laboratorium będą inne):

```
r-ab12
```

aws = AWS CLI: narzędzie chmury Amazonu

ZMIENIA SYSTEM

Utworzenie jednostki organizacyjnej na laboratorium: aws organizations create-organizational-unit

```
aws organizations create-organizational-unit --parent-id IdKorzenia --name Laboratorium
```

Jednostka `Laboratorium` dostanie później politykę SCP, która zamknie laboratorium w jednym regionie.

SCP = Service Connection Point: obiekt w Active Directory, po którym komputery domenowe znajdują swój tenant Entra ID przy dołączaniu hybrydowym · **aws** = AWS CLI: narzędzie chmury Amazonu

ZMIENIA SYSTEM

Utworzenie konta laboratorium w organizacji: aws organizations create-account

```
aws organizations create-account --email AdresEmail --account-name zielona-lab-aws
```

Adres e-mail musi być unikatowy w całym AWS, dlatego w Zielonej używamy aliasów skrzynki `aws-lab@zielona.example`. Konto powstaje w tle, a jego stan sprawdza `aws organizations describe-create-account-status --create-account-request-id IdZadania`.

AWS = Amazon Web Services: chmura Amazonu · **aws** = AWS CLI: narzędzie chmury Amazonu

ZMIENIA SYSTEM

Przeniesienie konta do jednostki Laboratorium: aws organizations move-account

```
aws organizations move-account --account-id IdKonta --source-parent-id IdKorzenia --destination-parent-id IdJednostki
```

aws = AWS CLI: narzędzie chmury Amazonu

ZMIENIA SYSTEM

Ścieżka w konsoli: włączenie IAM Identity Center

```
Konsola AWS → IAM Identity Center → Enable → Enable with AWS Organizations
Users → Add user → tomek
Permission sets → Create permission set → Predefined: AdministratorAccess, Session duration 4 hours
AWS accounts → zielona-lab-aws → Assign users or groups → tomek → AdministratorAccess
Settings → Authentication → Multi-factor authentication: Every time they sign in
```

Identity Center włączamy w regionie `eu-central-1`, bo jego region wybiera się raz. Użytkownik `tomek` loguje się przez portal dostępu z MFA i dostaje krótkotrwałe poświadczenia

do konta laboratorium.

◫◫ ZRZUT EKRANU DO UZUPEŁNIENIA

Zrzut do uzupełnienia: ekran IAM Identity Center → AWS accounts po przypisaniu, widoczne konto `zielona-lab-aws` w jednostce `Laboratorium` z użytkownikiem `tomek` i zestawem uprawnień `AdministratorAccess`.

◫◫ **IAM** = Identity and Access Management: zarządzanie tożsamościami i uprawnieniami (w AWS: użytkownicy, role, polityki) · **MFA** = Multi-Factor Authentication: logowanie drugim składnikiem, np. kodem z telefonu · **AWS** = Amazon Web Services: chmura Amazonu

◫◫ TYLKO ODCZYT

Konfiguracja profilu CLI logowanego przez Identity Center: aws configure sso

```
aws configure sso
```

Kreator pyta o nazwę sesji, adres portalu dostępu (z ekranu Settings w Identity Center), region `eu-central-1`, konto i zestaw uprawnień, a na końcu o nazwę profilu. W pliku `~/.aws/config` nie ląduje żaden klucz dostępu, tylko adres portalu i nazwa roli.

◫◫ **CLI** = Command Line Interface: praca przez wpisywanie komend w terminalu · **aws** = AWS CLI: narzędzie chmury Amazonu

◫◫ TYLKO ODCZYT

Zalogowanie profilu przed pracą: aws sso login

```
aws sso login --profile NazwaProfilu
```

Otwiera przeglądarkę z kodem urządzenia. Po zalogowaniu poświadczenia działają przez czas sesji z zestawu uprawnień, u nas cztery godziny.

◫◫ **aws** = AWS CLI: narzędzie chmury Amazonu

◫◫ TYLKO ODCZYT

Sprawdzenie, kim jesteśmy: aws sts get-caller-identity

```
aws sts get-caller-identity --profile NazwaProfilu
```

Przykład (kształt wyniku, wartości z laboratorium będą inne):

```
{
    "UserId": "AROAEXAMPLEID:tomek",
    "Account": "111122223333",
    "Arn": "arn:aws:sts::111122223333:assumed-
role/AWSReservedSSO_AdministratorAccess_0123456789abcdef/tomek"
}
```

W polu `Arn` widać rolę z przedrostkiem `AWSReservedSSO`, czyli logowanie przez Identity Center, a nie przez użytkownika IAM z kluczem.

IAM = Identity and Access Management: zarządzanie tożsamościami i uprawnieniami (w AWS: użytkownicy, role, polityki) · **aws** = AWS CLI: narzędzie chmury Amazonu

ZMIENIA SYSTEM

Założenie miesięcznego budżetu z alertem: aws budgets create-budget

```
aws budgets create-budget --account-id IdKonta --budget file://budzet.json --
notifications-with-subscribers file://powiadomienia.json
```

Budżet zakładamy na koncie zarządzającym, bo ono płaci za całą organizację. Dwa pliki są niżej. Budżet tylko ostrzega, nie wyłącza zasobów.

aws = AWS CLI: narzędzie chmury Amazonu

PLIK KONFIGURACYJNY

Plik budżetu: budzet.json

```
{
  "BudgetName": "lab-aws-miesiecznie",
  "BudgetLimit": { "Amount": "20", "Unit": "USD" },
  "TimeUnit": "MONTHLY",
  "BudgetType": "COST"
}
```

USD = dolar amerykański, waluta cenników AWS

PLIK KONFIGURACYJNY

Plik powiadomień budżetu: powiadomienia.json

```
[
  {
    "Notification": { "NotificationType": "ACTUAL", "ComparisonOperator":
"GREATER_THAN", "Threshold": 50, "ThresholdType": "PERCENTAGE" },
    "Subscribers": [ { "SubscriptionType": "EMAIL", "Address": "aws-lab@zielona.example"
} ]
  },
  {
    "Notification": { "NotificationType": "FORECASTED", "ComparisonOperator":
"GREATER_THAN", "Threshold": 100, "ThresholdType": "PERCENTAGE" },
    "Subscribers": [ { "SubscriptionType": "EMAIL", "Address": "aws-lab@zielona.example"
} ]
  }
]
```

Pierwsze powiadomienie przychodzi po wydaniu połowy budżetu, drugie wtedy, gdy prognoza na koniec miesiąca przekracza całość. Kwota 20 USD to nasza propozycja na miesiąc nauki.

USD = dolar amerykański, waluta cenników AWS

TYLKO ODCZYT

Sprawdzenie stref w regionie laboratorium: aws ec2 describe-availability-zones

```
aws ec2 describe-availability-zones --region eu-central-1 --query
"AvailabilityZones[].[ZoneName,ZoneId]" --output table
```

Przykład (kształt wyniku, wartości z laboratorium będą inne):

```
------------------------------------
|     DescribeAvailabilityZones     |
+----------------+-----------------+
```

```
|  eu-central-1a |  euc1-az2        |
|  eu-central-1b |  euc1-az3        |
|  eu-central-1c |  euc1-az1        |
+----------------+------------------+
```

Nazwy stref (`eu-central-1a`) są przypisywane do fizycznych stref losowo dla każdego konta, a identyfikatory (`euc1-az1`) są takie same dla wszystkich kont. Przy współpracy dwóch kont porównujemy identyfikatory.

aws = AWS CLI: narzędzie chmury Amazonu

TYLKO ODCZYT

Postawienie sieci laboratorium z szablonu: aws cloudformation deploy

```
aws cloudformation deploy --template-file lab-vpc.yaml --stack-name lab-vpc --parameter-overrides WlaczNat=false
```

Tworzy albo aktualizuje stos z pliku niżej. Z `WlaczNat=true` dochodzi brama NAT w jednej strefie, płatna za każdą godzinę, więc włączamy ją tylko na dni, w których ćwiczymy podsieci prywatne z wyjściem do internetu.

NAT = Network Address Translation: podmiana adresów IP na granicy sieci · **aws** = AWS CLI: narzędzie chmury Amazonu

TYLKO ODCZYT

Odczyt wyników stosu: aws cloudformation describe-stacks

```
aws cloudformation describe-stacks --stack-name lab-vpc --query "Stacks[0].Outputs" --output table
```

aws = AWS CLI: narzędzie chmury Amazonu

TYLKO ODCZYT

Usunięcie laboratorium po dniu pracy: aws cloudformation delete-stack

```
aws cloudformation delete-stack --stack-name lab-vpc
aws cloudformation wait stack-delete-complete --stack-name lab-vpc
```

Usuwa wszystko, co stos utworzył. Zasoby utworzone ręcznie w konsoli zostają i trzeba je znaleźć osobno, dlatego każdy ręcznie utworzony zasób oznaczamy tagiem `projekt=lab-aws`.

aws = AWS CLI: narzędzie chmury Amazonu

TYLKO ODCZYT

Wyszukanie zasobów laboratorium po tagu: aws resourcegroupstaggingapi get-resources

```
aws resourcegroupstaggingapi get-resources --tag-filters Key=projekt,Values=lab-aws --
query "ResourceTagMappingList[].ResourceARN" --output text
```

Pusta odpowiedź po sprzątaniu oznacza, że w tym regionie nie zostało nic oznaczonego. Polecenie widzi jeden region, więc przy ćwiczeniach z wieloma regionami powtarzamy je z `--region`.

aws = AWS CLI: narzędzie chmury Amazonu

PLIK KONFIGURACYJNY

Szablon sieci laboratorium: lab-vpc.yaml

```
# lab-vpc.yaml: VPC 10.60.0.0/16 w trzech strefach eu-central-1
AWSTemplateFormatVersion: "2010-09-09"
Description: Laboratorium SAA-C03 w Zielonej
Parameters:
  WlaczNat:
    Type: String
    AllowedValues: ["true", "false"]
    Default: "false"
Conditions:
  NatWlaczony:
    Fn::Equals: [Ref: WlaczNat, "true"]
Resources:
  Vpc:
```

```
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: 10.60.0.0/16
      EnableDnsSupport: true
      EnableDnsHostnames: true
      Tags:
        - {Key: Name, Value: vpc-lab-zielona}
        - {Key: projekt, Value: lab-aws}
  Igw:
    Type: AWS::EC2::InternetGateway
    Properties:
      Tags:
        - {Key: projekt, Value: lab-aws}
  IgwPodpiecie:
    Type: AWS::EC2::VPCGatewayAttachment
    Properties:
      VpcId: {Ref: Vpc}
      InternetGatewayId: {Ref: Igw}
  PubA: {Type: AWS::EC2::Subnet, Properties: {VpcId: {Ref: Vpc}, CidrBlock:
10.60.0.0/24, AvailabilityZone: eu-central-1a, MapPublicIpOnLaunch: true, Tags: [{Key:
Name, Value: pub-a}]}}
  PubB: {Type: AWS::EC2::Subnet, Properties: {VpcId: {Ref: Vpc}, CidrBlock:
10.60.1.0/24, AvailabilityZone: eu-central-1b, MapPublicIpOnLaunch: true, Tags: [{Key:
Name, Value: pub-b}]}}
  PubC: {Type: AWS::EC2::Subnet, Properties: {VpcId: {Ref: Vpc}, CidrBlock:
10.60.2.0/24, AvailabilityZone: eu-central-1c, MapPublicIpOnLaunch: true, Tags: [{Key:
Name, Value: pub-c}]}}
  AppA: {Type: AWS::EC2::Subnet, Properties: {VpcId: {Ref: Vpc}, CidrBlock:
10.60.10.0/24, AvailabilityZone: eu-central-1a, Tags: [{Key: Name, Value: app-a}]}}
  AppB: {Type: AWS::EC2::Subnet, Properties: {VpcId: {Ref: Vpc}, CidrBlock:
10.60.11.0/24, AvailabilityZone: eu-central-1b, Tags: [{Key: Name, Value: app-b}]}}
  AppC: {Type: AWS::EC2::Subnet, Properties: {VpcId: {Ref: Vpc}, CidrBlock:
10.60.12.0/24, AvailabilityZone: eu-central-1c, Tags: [{Key: Name, Value: app-c}]}}
  DbA: {Type: AWS::EC2::Subnet, Properties: {VpcId: {Ref: Vpc}, CidrBlock:
10.60.20.0/24, AvailabilityZone: eu-central-1a, Tags: [{Key: Name, Value: db-a}]}}
  DbB: {Type: AWS::EC2::Subnet, Properties: {VpcId: {Ref: Vpc}, CidrBlock:
10.60.21.0/24, AvailabilityZone: eu-central-1b, Tags: [{Key: Name, Value: db-b}]}}
  DbC: {Type: AWS::EC2::Subnet, Properties: {VpcId: {Ref: Vpc}, CidrBlock:
```

```
10.60.22.0/24, AvailabilityZone: eu-central-1c, Tags: [{Key: Name, Value: db-c}]}}
  TrasyPubliczne:
    Type: AWS::EC2::RouteTable
    Properties:
      VpcId: {Ref: Vpc}
  TrasaDoInternetu:
    Type: AWS::EC2::Route
    DependsOn: IgwPodpiecie
    Properties:
      RouteTableId: {Ref: TrasyPubliczne}
      DestinationCidrBlock: 0.0.0.0/0
      GatewayId: {Ref: Igw}
  PubAT: {Type: AWS::EC2::SubnetRouteTableAssociation, Properties: {SubnetId: {Ref:
PubA}, RouteTableId: {Ref: TrasyPubliczne}}}
  PubBT: {Type: AWS::EC2::SubnetRouteTableAssociation, Properties: {SubnetId: {Ref:
PubB}, RouteTableId: {Ref: TrasyPubliczne}}}
  PubCT: {Type: AWS::EC2::SubnetRouteTableAssociation, Properties: {SubnetId: {Ref:
PubC}, RouteTableId: {Ref: TrasyPubliczne}}}
  TrasyPrywatne:
    Type: AWS::EC2::RouteTable
    Properties:
      VpcId: {Ref: Vpc}
  AppAT: {Type: AWS::EC2::SubnetRouteTableAssociation, Properties: {SubnetId: {Ref:
AppA}, RouteTableId: {Ref: TrasyPrywatne}}}
  AppBT: {Type: AWS::EC2::SubnetRouteTableAssociation, Properties: {SubnetId: {Ref:
AppB}, RouteTableId: {Ref: TrasyPrywatne}}}
  AppCT: {Type: AWS::EC2::SubnetRouteTableAssociation, Properties: {SubnetId: {Ref:
AppC}, RouteTableId: {Ref: TrasyPrywatne}}}
  NatAdres:
    Type: AWS::EC2::EIP
    Condition: NatWlaczony
    Properties:
      Domain: vpc
  Nat:
    Type: AWS::EC2::NatGateway
    Condition: NatWlaczony
    Properties:
      SubnetId: {Ref: PubA}
```

```
      AllocationId:
        Fn::GetAtt: [NatAdres, AllocationId]
  TrasaPrzezNat:
    Type: AWS::EC2::Route
    Condition: NatWlaczony
    Properties:
      RouteTableId: {Ref: TrasyPrywatne}
      DestinationCidrBlock: 0.0.0.0/0
      NatGatewayId: {Ref: Nat}
Outputs:
  VpcId: {Value: {Ref: Vpc}}
  PodsieciPubliczne:
    Value:
      Fn::Join: [",", [{Ref: PubA}, {Ref: PubB}, {Ref: PubC}]]
  PodsieciAplikacji:
    Value:
      Fn::Join: [",", [{Ref: AppA}, {Ref: AppB}, {Ref: AppC}]]
  PodsieciBaz:
    Value:
      Fn::Join: [",", [{Ref: DbA}, {Ref: DbB}, {Ref: DbC}]]
```

Funkcje CloudFormation zapisujemy w pełnej postaci (`Fn::Join`, `Ref:`), a nie skrótem z wykrzyknikiem, żeby plik czytał każdy parser YAML. Podsieci baz nie mają trasy do internetu w ogóle. Jedna brama NAT w jednej strefie to oszczędność laboratorium, a nie wzór dla produkcji: w produkcji stawiamy bramę NAT w każdej strefie albo bramę regionalną (szczegóły na stronie o sieci).

YAML = YAML Ain't Markup Language: format plików konfiguracyjnych oparty na wcięciach · **NAT** = Network Address Translation: podmiana adresów IP na granicy sieci · **SAA-C03** = kod wersji egzaminu AWS Solutions Architect Associate zdawanej od 30 sierpnia 2022, aktualnej w październiku 2026 · **AWS** = Amazon Web Services: chmura Amazonu · **EC2** = Elastic Compute Cloud: usługa AWS wynajmująca gotowe maszyny wirtualne na godziny · **VPC** = Virtual Private Cloud: prywatna sieć wirtualna klienta w chmurze AWS, w której stawiamy maszyny i zakładamy tablice tras

Z życia wzięte

Z ŻYCIA WZIĘTE

> Czwartek, 21:40. Tomek ćwiczy bramę NAT, kończy o 22:00 i zamyka laptopa. W poniedziałek budżet przysyła e-mail o przekroczeniu połowy kwoty, bo brama NAT i nieprzypisany adres publiczny liczyły się przez cały weekend. Tomek uruchamia `get-resources` z tagiem `projekt=lab-aws`, znajduje adres, który zrobił ręcznie poza stosem, i od tego dnia kończy każdy dzień poleceniem `delete-stack` i sprawdzeniem po tagu.

Plan 28 dni

Plan ma pięć etapów: dwa dni konta i laboratorium, siedem dni bezpieczeństwa (30 procent egzaminu), siedem dni odporności (26 procent), cztery dni wydajności (24 procent), cztery dni kosztów (20 procent) i trzy dni egzaminu. Dni 7 i 14 to powtórki z mini sprawdzianem.



Rysunek 1. Plan na 28 dni: laboratorium, cztery domeny w kolejności wag, powtórki w dniach 7, 14 i 21, egzamin próbny w dniu 26 i egzamin w dniu 28 (kliknięcie otwiera pełny rozmiar)

Termin egzaminu rezerwujemy w dniu pierwszym, a dostosowanie ESL zgłaszamy w koncie AWS Certification przed rezerwacją, bo bez niego nie dostaniemy dodatkowych trzydziestu minut. Gdy termin wypada później, dni 26 i 27 powtarzamy z nowym zestawem pytań, a nie dokładamy nowych tematów.

Dzień	Temat	Zadania przewodnika	Strona	Laboratorium	Czas

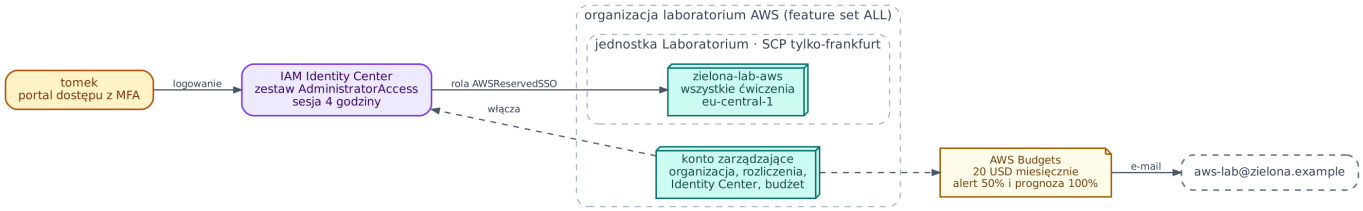
1	Konto, MFA roota, Organizations, Identity Center, budżet	1.1	ta strona	organizacja, konto lab, `tomek` z MFA, budżet 20 USD	2 h
2	CLI, CloudShell, CloudFormation, sprzątanie	2.2 (automatyzacja)	ta strona	stos `lab-vpc`, usunięcie, sprawdzenie po tagu	2 h
3	IAM: polityki, logika oceny	1.1	Bezpieczne, część 1	polityka z warunkiem, symulacja	1,5 h
4	Role, STS, dostęp między kontami, federacja	1.1	Bezpieczne, część 1	rola dla EC2, rola między kontami	1,5 h
5	Organizations, SCP, Control Tower, polityki zasobów	1.1	Bezpieczne, część 1	SCP blokujący regiony poza Frankfurtem	1,5 h
6	VPC: podsieci, trasy, NAT, SG, NACL	1.2	Bezpieczne, część 2	trzy warstwy podsieci, SG z odwołaniem do SG	2 h
7	Powtórka i 15 pytań z ćwiczeń dni 3 do 6	1.1, 1.2	strony dni 3 do 6	dziennik błędów	1,5 h
8	Endpointy, PrivateLink, VPN, Direct Connect, WAF, Shield, GuardDuty, Macie, Cognito	1.2	Bezpieczne, część 2	endpoint bramowy S3	1,5 h
9	KMS, ACM, Secrets Manager, Parameter Store	1.3	Bezpieczne, część 2	klucz KMS z rotacją, sekret bazy	1,5 h
10	ELB: ALB, NLB, GWLB	2.1, 2.2	Odporne, część 1	ALB z grupą docelową	2 h
11	EC2 Auto Scaling	2.1, 3.2	Odporne, część 1	grupa w trzech strefach, śledzenie celu	2 h
12	Route 53 i polityki routingu	2.2	Odporne, część 1	strefa prywatna, failover z kontrolą zdrowia	1,5 h

13	SQS, SNS, EventBridge, Step Functions, API Gateway	2.1	Odporne, część 1	kolejka z kolejką martwych komunikatów	1,5 h
14	Powtórka i 20 pytań z dni 8 do 13	1.2, 1.3, 2.1, 2.2	strony dni 8 do 13	dziennik błędów	1,5 h
15	RDS Multi-AZ, repliki odczytu, RDS Proxy	2.2, 3.3	Odporne, część 2	MariaDB Multi-AZ, wymuszone przełączenie	2 h
16	Aurora, Aurora Global Database, DynamoDB global tables	2.2, 3.3	Odporne, część 2	klaster Aurora (opcjonalnie, płatny)	1,5 h
17	AWS Backup, strategie DR, RTO i RPO	2.2, 1.3	Odporne, część 2	plan kopii, odtworzenie	1,5 h
18	EC2: rodziny, grupy rozmieszczenia, Lambda	3.2	Wydajne	funkcja Lambda z kolejki	1,5 h
19	ECS, Fargate, EKS, kontenery	2.1, 3.2	Wydajne	usługa Fargate za ALB	2 h
20	EBS, EFS, FSx, wydajność S3	3.1	Wydajne	gp3 z IOPS, EFS w trzech strefach	1,5 h
21	DynamoDB, ElastiCache, CloudFront, Global Accelerator, Kinesis, Glue, Athena; powtórka	3.3, 3.4, 3.5	Wydajne	tabela na żądanie, CloudFront przed S3	2 h
22	Modele zakupu EC2, Savings Plans, Spot	4.2	Koszty	rekomendacja Savings Plans, ceny Spot	1,5 h
23	Klasy S3, cykl życia, przesył danych	4.1	Koszty	reguła cyklu życia	1,5 h
24	Koszty sieci: NAT, endpointy, transfer	4.4	Koszty	porównanie NAT i endpointu	1,5 h
25	Cost Explorer, Budgets, CUR, Compute Optimizer, tagi; koszty baz	4.1 do 4.4	Koszty	raport kosztów po usłudze i tagu	1,5 h

26	Egzamin próbny 65 pytań na czas	wszystkie	Próbny egzamin	130 minut bez notatek	2,5 h
27	Analiza błędów, oficjalne pytania AWS Skill Builder	wszystkie	Próbny egzamin	dziennik błędów	2 h
28	Egzamin	wszystkie	Próbny egzamin	egzamin	ok. 2,5 h

Laboratorium

Laboratorium to dwa konta w osobnej organizacji: konto zarządzające, na którym są tylko organizacja, Identity Center, rozliczenia i budżet, oraz konto `zielona-lab-aws`, na którym robimy wszystkie ćwiczenia. Konto zarządzające to odpowiednik szafy z kluczami: nikt na nim nie uruchamia maszyn.



Rysunek 2. Konta laboratorium: organizacja z kontem zarządzającym, jednostka Laboratorium z kontem ćwiczeń, IAM Identity Center z użytkownikiem tomek, SCP na jednostce i budżet z alertem e-mail (kliknięcie otwiera pełny rozmiar)

Konto i Free Tier w 2025 i 2026 roku

Od 15 lipca 2025 roku nowe konto AWS wybiera przy zakładaniu plan darmowy albo płatny. Oba dostają 100 USD kredytów na start i do 100 USD więcej za wykonanie zadań w wybranych usługach. Plan darmowy kończy się po sześciu miesiącach albo po wyczerpaniu kredytów, a wtedy konto zamyka się samo i AWS trzyma dane 90 dni na przejście na plan płatny.

Sprawa	Plan darmowy	Plan płatny
Usługi	wybrane, bez Savings Plans, Reserved Instances i części ofert z Marketplace	wszystkie
Po wyczerpaniu kredytów	konto się zamyka	płacimy według cennika
AWS Organizations	dołączenie samo przełącza konto na plan płatny	bez ograniczeń

Sprawa	Plan darmowy	Plan płatny
Ponad 30 usług zawsze darmowych w limitach	tak	tak

Wniosek dla laboratorium: skoro używamy Organizations, konto zarządzające i tak będzie na planie płatnym, a kredyty z rejestracji zostają na nim do wykorzystania. Czy konta tworzone w organizacji poleceniem `create-account` dostają własne kredyty, AWS na stronach o planach nie pisze, więc sprawdzamy to po utworzeniu poleceniem `get-account-plan-state` (do potwierdzenia). Konta założone przed 15 lipca 2025 roku działają według poprzednich zasad Free Tier.

Pierwsza godzina na koncie zarządzającym

1. Logujemy się na użytkownika głównego (root), włączamy MFA i zapisujemy kody odzyskiwania w Vaultwardenie Zielonej.
2. Nie tworzymy kluczy dostępu dla użytkownika głównego. Jeżeli kreator konta je utworzył, usuwamy.
3. Zakładamy organizację, jednostkę `Laboratorium` i konto `zielona-lab-aws` poleceniami z Fiszki albo w konsoli.
4. Włączamy IAM Identity Center, zakładamy użytkownika `tomek` z MFA i przypisujemy mu zestaw `AdministratorAccess` na koncie laboratorium. Na koncie zarządzającym `tomek` nie dostaje nic.
5. Zakładamy budżet 20 USD z dwoma powiadomieniami.
6. Od tej chwili konta użytkownika głównego używamy tylko do zadań, które go wymagają, a pracujemy przez portal Identity Center.

> **Pojęcie: IAM Identity Center.** Usługa logowania jednokrotnego do wielu kont AWS. Użytkownik loguje się raz, wybiera konto i zestaw uprawnień, a AWS wydaje mu poświadczenia ważne kilka godzin. Dawna nazwa to AWS SSO, stąd polecenia `aws sso` i `aws configure sso`.

Dlaczego nie klucze dostępu: klucz użytkownika IAM działa, dopóki ktoś go nie usunie, więc wyciek z pliku `~/.aws/credentials` albo z historii powłoki daje dostęp na długo. Poświadczenia z Identity Center wygasają same. Ta różnica wraca na egzaminie w każdym pytaniu o bezpieczny dostęp ludzi.

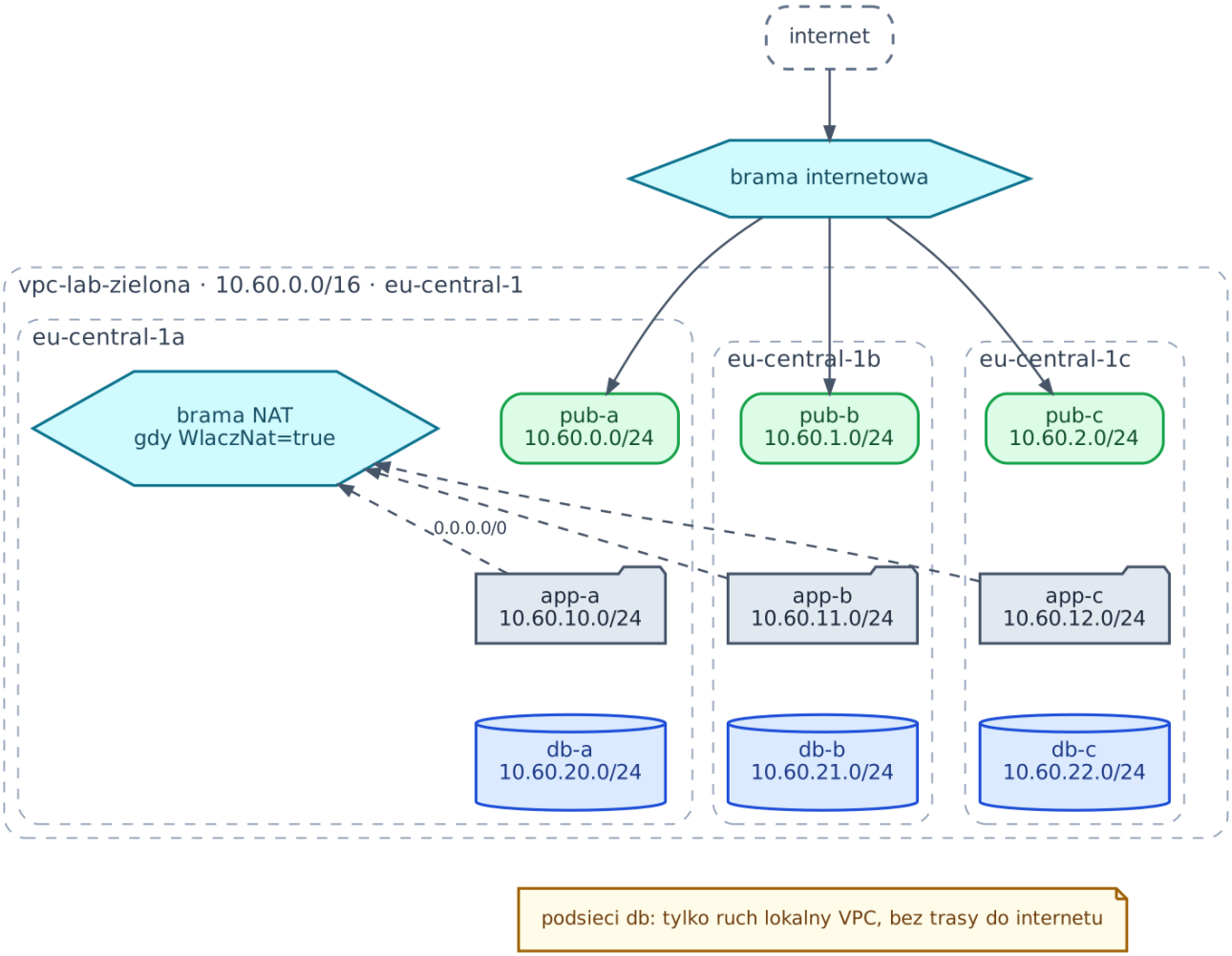
AWS CLI i CloudShell

>

> **Pojęcie: AWS CloudShell.** Powłoka w przeglądarce, otwierana z konsoli AWS, z gotowym AWS CLI i poświadczeniami zalogowanego użytkownika. Ma 1 GB trwałego miejsca w katalogu domowym w każdym regionie, bez opłat, a dane z regionu, w którym nie używaliśmy jej przez 120 dni, są usuwane.

CloudShell wystarcza do całego kursu: nie trzeba niczego instalować, a plik `lab-vpc.yaml` wgrywamy przez menu Actions → Upload file. Na własnym komputerze instalujemy AWS CLI v2 według strony AWS CLI: instalacja, profile, poświadczenia, ale zamiast `aws configure` z kluczami używamy `aws configure sso`. Sesja CloudShell kończy się po 20 do 30 minutach bezczynności.

Sieć laboratorium



Rysunek 3. VPC laboratorium 10.60.0.0/16 w eu-central-1: trzy strefy, w każdej podsieć publiczna, aplikacji i baz, brama internetowa i opcjonalna brama NAT w strefie a (kliknięcie otwiera pełny rozmiar)

Warstwa	eu-central-1a	eu-central-1b	eu-central-1c	Trasa domyślna
publiczna (`pub`)	10.60.0.0/24	10.60.1.0/24	10.60.2.0/24	brama internetowa
aplikacji (`app`)	10.60.10.0/24	10.60.11.0/24	10.60.12.0/24	brama NAT, gdy `WlaczNat=true`
baz (`db`)	10.60.20.0/24	10.60.21.0/24	10.60.22.0/24	brak, tylko ruch lokalny VPC

Sieć 10.60.0.0/16 nie występuje nigdzie indziej w bazie. AWS rezerwuje w każdej podsieci pięć adresów (pierwsze cztery i ostatni), więc podsieć /24 ma 251 adresów dla nas. Wolne zakresy 10.60.3.0 do 10.60.9.255 i od 10.60.23.0 zostają na ćwiczenia z kolejnych stron.

Sprzątanie i koszty

Każdy dzień kończymy w tej samej kolejności: usuwamy zasoby ćwiczenia (bazy, równoważniki, grupy Auto Scaling), potem stos `lab-vpc`, a na końcu sprawdzamy tagi. Najczęstsze źródła rachunku po nauce to brama NAT, równoważnik obciążenia, instancja bazy, nieprzypisany adres IPv4 (płatny jak każdy publiczny adres IPv4) i migawki dysków.

Szablon CloudFormation jest naszym narzędziem do odtwarzania laboratorium. Ten sam szablon można przepisać na Terraform, opisany w książce o automatyzacji, ale egzamin pyta o CloudFormation jako usługę AWS do infrastruktury jako kodu, więc w kursie trzymamy się jej.

Jak pracujemy codziennie

Dzień składa się z trzech części: około dwudziestu minut czytania strony domeny i dokumentacji AWS, godziny w laboratorium i dwudziestu minut na pytania z ćwiczeń, odpowiadane bez zaglądania. Do każdej usługi zapisujemy w dzienniku jedno zdanie: kiedy jest odpowiedzią, a kiedy pułapką.

Egzamin jest po angielsku, więc nazwy usług i pojęć uczymy się w oryginale. W tej książce przy każdym pojęciu stoi nazwa angielska, a w pytaniach próbnych zostawiamy angielskie słowa kluczowe, takie jak „most cost-effective” i „least operational overhead”.

Częste błędy

Objaw	Przyczyna	Co robimy

`create-account` kończy się błędem o adresie e-mail	adres jest już użyty przez inne konto AWS	alias skrzynki, na przykład `aws-lab+2@zielona.example`, jeśli serwer poczty obsługuje znak plus
Część usług jest niedostępna albo przyciski są szare	konto na planie darmowym	przejście na plan płatny albo dołączenie do organizacji
`aws sso login` otwiera stronę z błędem	zły adres portalu albo region Identity Center w profilu	`aws configure sso` jeszcze raz, region `eu-central-1`
`delete-stack` kończy się stanem `DELETE_FAILED`	zasób spoza stosu korzysta z podsieci, na przykład ręcznie utworzona instancja	usuwamy go i ponawiamy `delete-stack`
Rachunek mimo usuniętego stosu	zasoby w innym regionie albo bez tagu	`get-resources` w każdym użytym regionie, Cost Explorer według usług
Budżet nie wysłał e-maila	nie potwierdziliśmy subskrypcji albo alert dotyczy prognozy, której jeszcze nie ma	sprawdzamy powiadomienia w konsoli Budgets

□□ Ćwiczenia

1. Dlaczego budżet zakładamy na koncie zarządzającym, a nie na koncie laboratorium?

Odpowiedź

W organizacji rachunek za wszystkie konta płaci konto zarządzające (konsolidacja rozliczeń), a budżet na nim widzi koszty całej organizacji. Budżet na koncie laboratorium widziałby tylko to konto. Można mieć oba, ale ten na koncie zarządzającym chroni przed zapomnianym zasobem na dowolnym koncie.

1. Tomek w CloudShell widzi pustą listę instancji, chociaż w konsoli widział je rano. Co sprawdzamy najpierw?

Odpowiedź

Region. CloudShell działa w regionie wybranym w konsoli, a polecenia bez `--region` pytają tylko ten region. Sprawdzamy `aws configure get region` i prawy górny róg konsoli, a potem powtarzamy z `--region eu-central-1`.

1. W pytaniu egzaminacyjnym firma chce, żeby administratorzy nie mieli stałych kluczy dostępu, a logowali się kontem z firmowego katalogu do kilkunastu kont AWS. Która usługa jest odpowiedzią?

Odpowiedź

IAM Identity Center z dostawcą tożsamości (na przykład Active Directory przez AWS Directory Service albo zewnętrzny dostawca SAML) i zestawami uprawnień przypisanymi do kont. Użytkownicy IAM z kluczami w każdym koncie są odpowiedzią złą, bo dają stałe poświadczenia i rozproszone zarządzanie.

1. Ile adresów dla instancji ma podsieć 10.60.20.0/24 i które adresy zabiera AWS?

Odpowiedź

1. AWS rezerwuje 10.60.20.0 (adres sieci), .1 (router VPC), .2 (DNS), .3 (rezerwa na przyszłość) i .255 (adres rozgłoszeniowy, nieobsługiwany, ale zarezerwowany).

Powiązane: AWS Solutions Architect Associate: od kolejki SQS do egzaminu SAA-C03 · AWS CLI: instalacja, profile, poświadczenia · Chmura: pojęcia, które trzeba znać zanim cokolwiek uruchomimy

Źródła: docs.aws.amazon.com/awsaccountbilling (Choosing a plan, Free Tier); aws.amazon.com/about-aws/whats-new/2025/07 (Free Tier, 200 USD kredytów i plan darmowy na sześć miesięcy); docs.aws.amazon.com/cloudshell (limity: 1 GB na region, 120 dni, sesje); dokumentacja AWS CLI v2 (`organizations`, `budgets`, `cloudformation`, `configure sso`, `sso login`, `freetier`); dokumentacja Amazon VPC (adresy zarezerwowane w podsieci, identyfikatory stref); dokumentacja IAM Identity Center. Parametry poleceń sprawdzone z modelem usług AWS CLI (botocore 1.43) 19 października 2026.

Revision #3
Created 2026-10-18 23:00:31 UTC by Tomasz Konieczny
Updated 2026-10-26 14:27:28 UTC by Tomasz Konieczny