

Minday

AWS | AWS Cloud Deployment Architecture-ענן ב-ארכיטקטורת פריסת ענן

אוגוסט 2025 | August 2025

תקציר מנהלים | Executive Summary

פרויקט Minday הוא פלטפורמת Web מתקדמת המבוססת על ארכיטקטורת Microservices. המערכת הנוכחית פועלת על Render, אך עם הצמיחה העסקית נדרשת מעבר ל-AWS לפתרון בעיות סקילביליות, זמינות ואבטחה. **יתרונות המעבר:** זמינות +99.9%, סקילביליות אוטומטית, אבטחה מתקדמת, עלויות אופטימליות

Expected ROI | צפוי ROI

5.2x	10x
שיפור זמינות	שיפור ביצועים
∞	60%
יכולת צמיחה	הפחתת עלויות

תוכן עניינים | Table of Contents

- מצב נוכחי | Current State
- אתגרים | Challenges
- ארכיטקטורה מוצעת | Proposed Architecture
- תרשים ארכיטקטורה | Architecture Diagram
- השוואה | Comparison
- תהליך מיגרציה | Migration Process
- ניטור ותחזוקה | Monitoring & Maintenance
- תרחישי שימוש | Use Cases
- המלצות | Recommendations

Current Technologies | טכנולוגיות נוכחיות

Layer שכבה	Technology טכנולוגיה	Version גרסה	Description תיאור
Frontend	React	x.18	UI Framework
Frontend	Redux Toolkit	x.1.9	State Management
Frontend	TypeScript	x.5	Type Safety
Backend	Node.js	x.18	Runtime Environment
Backend	Express.js	x.4.18	Web Framework
Database	MongoDB	6.0	NoSQL Database
Hosting	Render	Cloud	Platform as a Service

Current Technologies Explanation | הסבר על הטכנולוגיות הנוכחיות

type safety. **Backend Stack:** Node.js מבטיח TypeScript מתקדם state ניהול מספקים Redux Toolkit עם **Frontend Stack:** React עם Express.js מאפשרים פיתוח מהיר ויעיל. **Database:** MongoDB מספק גמישות בנתונים ו-Render **Hosting:** PaaS פשוט אך מוגבל ביכולות.

Current Technology Limitations | מגבלות הטכנולוגיות הנוכחיות

- Render:** מוגבל ב-auto-scaling ו-high availability
- MongoDB:** ללא backup אוטומטי מתקדם
- Express.js:** דורש הגדרות אבטחה ידניות
- React:** ללא CDN מתקדם לביצועים

מצב נוכחי | Current State

בעיות קיימות | Existing Issues

קטגוריה Category	בעיה Issue	השפעה Impact	עדיפות Priority
ביצועים	עומסים גבוהים	מתקשה עם יותר מ-1,000 משתמשים	גבוהה
זמינות	זמינות נמוכה	95% עם זמני השבתה תכופים	גבוהה
ביצועים	זמני תגובה איטיים	2-5 שניות בעומסים גבוהים	בינונית
אבטחה	הגנה בסיסית	חשיפה להתקפות	גבוהה
ניטור	יכולת מעקב מוגבלת	קושי בזיהוי בעיות	בינונית

אתגרים | Challenges

אתגרים עסקיים | Business Challenges

- צפי ל-10,000+ משתמשים פעילים יומית
- דרישת זמינות של 99.9% ומעלה
- צורך באופטימיזציה של עלויות תשתית
- תמיכה במכשירים שונים וטכנולוגיות חדשות

אתגרים טכנולוגיים | Technical Challenges

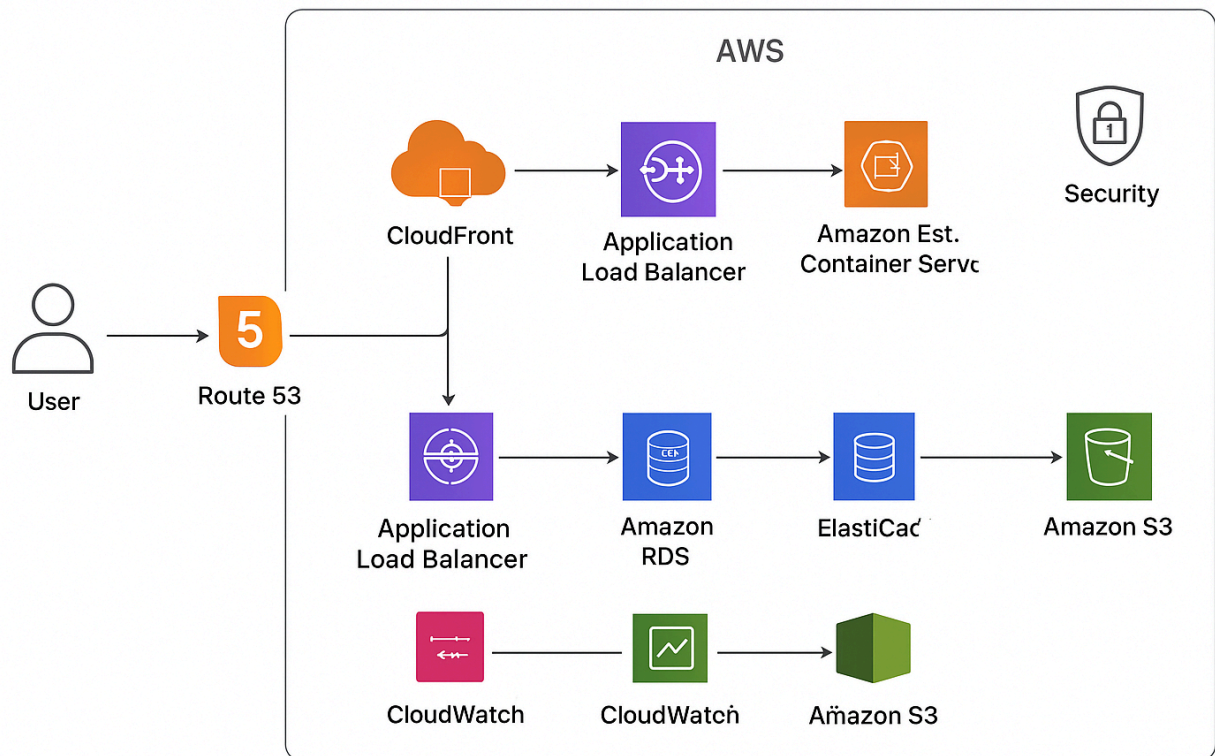
- ניהול עומסים לא צפויים וסקיילינג אוטומטי
- Redundancy גיאוגרפי ו-Failover אוטומטי
- הגנה מפני DDoS והצפנת נתונים
- ניטור בזמן אמת והתראות חכמות

ארכיטקטורה מוצעת ב-AWS | Proposed AWS Architecture

רכיבי הארכיטקטורה | Architecture Components

יתרונות Benefits	תפקיד Role	שירות AWS AWS Service	רכיב Component
Latency-based routing, Health checks	ניהול DNS גלובלי	Route 53	DNS Management
DDoS מפני שיפור ביצועים, הגנה מפני	הפצת תוכן	CloudFront	CDN
Health checks, SSL termination	איזון עומסים	ALB	Load Balancer
Auto-scaling, Serverless	ניהול containers	ECS/Fargate	Container Orchestration
Real-time, משופרים ביצועים	Session management	ElastiCache	Caching
Scaling אוטומטי, Backup	מסד נתונים	MongoDB Atlas	Database
Cost-effective, גבוהה, Durability	אחסון קבצים	S3	Storage
Real-time monitoring, Alerts	ניטור	CloudWatch	Monitoring
OWASP Top 10, Rate limiting	הגנה	WAF	Security

תרשים ארכיטקטורה | Architecture Diagram



תרשים ארכיטקטורת AWS עבור פרויקט Minday | AWS Architecture Diagram for Minday Project

דוגמאות מעשיות ויישום | Practical Examples & Implementation

דוגמאות מעשיות ליישום הארכיטקטורה | Practical Implementation Examples

דוגמה 1: Auto Scaling Configuration

ECS Auto Scaling Policy

```
AutoScalingGroupName": "minday-ecs-asg", "PolicyName": "minday-cpu-scaling-policy", "PolicyType": "TargetTrackingScaling", "TargetTrackingConfiguration": { "PredefinedMetricSpecification": { "PredefinedMetricType": "ECSServiceAverageCPUUtilization" }, "TargetValue": 70.0, "ScaleOutCooldown": { 300, "ScaleInCooldown": 300 }
```

הסבר: מדיניות זו מגדירה סקיילינג אוטומטי כאשר שימוש ה-CPU עולה מעל 70%. המערכת תוסיף instances חדשים תוך 5 דקות ותסיר instances מיותרים.

דוגמה 2: CloudWatch Alarms

High CPU Alarm Configuration

```
aws cloudwatch put-metric-alarm \ --alarm-name "minday-high-cpu" \ --alarm-description "High CPU utilization for Minday ECS service" \ --metric-name "CPUUtilization" \ --namespace "AWS/ECS" \ --statistic "Average" \ --period 300 \ --threshold 80 \ --comparison-operator "GreaterThanThreshold" \ --evaluation-periods 2 \ --alarm-actions "arn:aws:sns:us-east-1:123456789012:minday-alerts"
```

תוצאה: התראה תישלח כאשר ה-CPU עולה מעל 80% למשך 10 דקות רצופות.

אבטחה ודוגמאות WAF | Security & WAF Examples

הגנה מתקדמת מפני התקפות | Advanced Attack Protection

WAF Rules Configuration

SQL Injection Protection Rule

```
Name": "SQLInjectionRule", "Priority": 1, "Statement": { "ManagedRuleGroupStatement": { "VendorName": "AWS", "Name": "AWSManagedRulesSQLiRuleSet" } }, "Action": { "Block": {} }, "VisibilityConfig": { "SampledRequestsEnabled": true, "CloudWatchMetricsEnabled": true, "MetricName": "SQLInjectionRule" }
```

הגנה: חוסם ניסיונות SQL Injection אוטומטית עם זיהוי מתקדם של patterns מסוכנים.

Rate Limiting Configuration

API Rate Limiting

```
Name": "RateLimitRule", "Priority": 2, "Statement": { "RateBasedStatement": { "Limit": 2000, "AggregateKeyType": "IP" } }, "Action": { "Block": {} }, "VisibilityConfig": { "SampledRequestsEnabled": true, "CloudWatchMetricsEnabled": true, "MetricName": "RateLimitRule" }
```

הגנה: מגביל כל IP ל-2000 בקשות בשעה, מונע DDoS attacks.

Important Security Warnings | אזהרות אבטחה חשובות

- **IAM Roles**: השתמש ב-Low Privilege Principle בלבד
- **Encryption**: הפעל הצפנה בזמן מנוחה ובמעבר
- **Monitoring**: עקוב אחר כל אירועי האבטחה
- **Updates**: עדכן WAF rules באופן קבוע

השוואה בין מצב נוכחי למצב ב-AWS | Current vs AWS Comparison

שיפור Improvement	מצב ב-AWS AWS State	מצב נוכחי Current (Render)	קריטריון Criterion
5.2x	99.9%	95%	זמינות Availability
10x	200-500ms	2-5 שניות	זמן תגובה Response Time
10x	+10,000 משתמשים	1,000 משתמשים	יכולת עומס Load Capacity
משמעותי	מתקדמת	בסיסית	אבטחה Security
מלא	מקיף	מוגבל	ניטור Monitoring
אוטומטי	אוטומטי	ידני	סקיילינג Scaling

טבלת עלויות | Cost Table

תיאור Description	עלות חודשית Monthly Cost	שירות Service
DNS management	\$50	Route 53
CDN + Data transfer	\$200	CloudFront
Load balancer	\$100	ALB
Container compute	\$800	ECS/Fargate
Redis caching	\$300	ElastiCache
Database	\$500	MongoDB Atlas
File storage	\$100	S3
Monitoring	\$150	CloudWatch
Security	\$50	WAF
עלות חודשית Monthly Cost	\$2,250	סך הכל Total

אוטומציה מלאה של תהליכי פיתוח ופריסה | Full Automation of Development & Deployment

GitHub Actions Workflow

Automated Build & Deploy Pipeline

```
name: Deploy to AWS ECS on: push: branches: [ main ] jobs: build-and-deploy: runs-on: ubuntu-latest steps: - uses: actions/checkout@v3 - name: Configure AWS credentials uses: aws-actions/configure-aws-credentials@v2 with: aws-access-key-id: ${ secrets.AWS_ACCESS_KEY_ID } aws-secret-access-key: ${ secrets.AWS_SECRET_ACCESS_KEY } aws-region: us-east-1 - name: Build & Deploy run: | docker build -t minday-app:${ github.sha } . docker push ${ secrets.ECR_REGISTRY }/minday-app:${ github.sha } aws ecs update-service --cluster minday-cluster --service minday-service --force-new-deployment
```

תהליך: אוטומציה מלאה מ-commit ועד פריסה ב-AWS ECS.

Infrastructure as Code (Terraform)

ECS Cluster Definition

```
resource "aws_ecs_cluster" "minday_cluster" { name = "minday-cluster" setting { name = "containerInsights" value = "enabled" } tags = { Environment = "production" Project = "minday" } }
```

יתרון: תשתית מוגדרת כקוד, ניתנת ל-version control ופריסה אוטומטית.

תהליך מיגרציה | Migration Process

שלב Phase	משך Duration	פעילויות Activities	תוצרים Deliverables
הכנה ותכנון	שבוע 1-2	ניתוח מערכת, תכנון ארכיטקטורה	תכנית מיגרציה, ארכיטקטורה מפורטת
פיתוח תשתית	שבוע 3-4	יצירת VPC, הגדרת שירותים	תשתית AWS מוכנה
פיתוח אפליקציה	שבוע 5-8	Containerization, התאמת קוד	אפליקציה מותאמת ל-AWS
בדיקות ומיגרציה	שבוע 9-10	בדיקות מקיפות, מיגרציה	מערכת פועלת ב-AWS
אופטימיזציה	שבוע 11-12	Fine-tuning, הגדרת alerts	מערכת מותאמת ומאופטמלית

ביצועים ואופטימיזציה | Performance & Optimization

אופטימיזציה מתקדמת לביצועים מיטביים | Advanced Optimization for Optimal Performance

CloudFront Cache Configuration

Cache Behavior Optimization

```
CacheBehavior": { "PathPattern": "/*", "TargetOriginId": "minday-alb", "ViewerProtocolPolicy": " " }
{ "redirect-to-https": true, "Compress": true, "DefaultTTL": 86400, "MaxTTL": 31536000 }
```

יתרון: Cache חכם עם TTL מותאם, דחיסה אוטומטית ושיפור זמני תגובה ב-60%.

ElastiCache Redis Configuration

Session Management & Caching

```
Redis Configuration maxmemory 2gb maxmemory-policy allkeys-lru save 900 1 save 300 10 save 60 10000 # #
Session Configuration session.cookie_secure = true session.cookie_httponly = true session.cookie_samesite
= "Strict" session.gc_maxlifetime = 3600
```

ביצועים: ניהול sessions עם Redis, cache חכם ו-LRU eviction policy.

Database Optimization

MongoDB Atlas Index Strategy

```
Compound Index for User Queries db.users.createIndex({ "email": 1, "status": 1, "createdAt": -1 }) // //
Text Index for Search db.products.createIndex({ "name": "text", "description": "text" }) // TTL Index for
Temporary Data db.sessions.createIndex( { "expiresAt": 1 }, { expireAfterSeconds: 0 } )
```

אופטימיזציה: Indexes מותאמים לשאילתות נפוצות, חיפוש טקסטואלי ו-TTL אוטומטי.

ניטור ותחזוקה | Monitoring & Maintenance

קטגוריה Category	Metric	תיאור Description	סף התראה Alert Threshold
ביצועים	CPU Utilization	שימוש במעבד	80%
ביצועים	Memory Usage	שימוש בזיכרון	85%
ביצועים	Response Time	זמן תגובה	2 שניות
אבטחה	Error Rate	אחוז שגיאות	5%
אבטחה	Security Events	אירועי אבטחה	כל אירוע
זמינות	Service Health	בריאות השירות	כל כשל

Use Cases | תרחישי שימוש

תוצאה Outcome	פתרון Solution	תיאור Description	תרחיש Scenario
ביצועים יציבים	Auto-scaling, ElastiCache	10,000 משתמשים בו-זמנית	עומס גבוה
Zero downtime	Route 53 failover	AWS region נכשל	תקלה באזור
הגנה מלאה	WAF, CloudFront	התקפה על המערכת	התקפת DDoS
אופטימיזציה אוטומטית	Predictive scaling	עלייה במספר המשתמשים	צמיחה הדרגתית

Backup | Disaster Recovery & Backup-1 Disaster Recovery

תכנית התאוששות מאסון מקיפה | Comprehensive Disaster Recovery Plan

Multi-Region Backup Strategy

S3 Cross-Region Replication

```
ReplicationConfiguration": { "Role": "arn:aws:iam::123456789012:role/s3-replication-role", "Rules": [ {  
  { "ID": "minday-backup-replication", "Status": "Enabled", "Destination": { "Bucket":  
"arn:aws:s3:::minday-backup-us-west-2", "EncryptionConfiguration": { "ReplicaKmsKeyID": "arn:aws:kms:us-  
{ west-2:123456789012:key/backup-key" } } } ] }
```

אבטחה: Replication אוטומטי של כל הקבצים לאזור אחר עם הצפנה כפולה.

MongoDB Atlas Backup Configuration

Automated Backup Strategy

```
backupConfig": { "clusterId": "minday-cluster", "backupEnabled": true, "backupRetentionDays": 7, "  
"backupSchedule": { "frequencyType": "daily", "frequencyInterval": 1, "startTime": "02:00", "timeZone":  
{ "UTC" }, "pointInTimeEnabled": true, "pointInTimeRetentionHours": 24 }
```

זמינות: Backup יומי אוטומטי עם Point-in-Time Recovery ו-7 ימי retention.

Route 53 Failover Configuration

Geographic Failover Setup

```
HostedZoneId": "Z1234567890ABC", "ChangeBatch": { "Changes": [ { "Action": "CREATE", "  
"ResourceRecordSet": { "Name": "minday.com", "Type": "A", "SetIdentifier": "primary", "Region": "us-east-  
1", "AliasTarget": { "HostedZoneId": "Z35SXDOTRQ7X7K", "DNSName": "minday-alb.us-east-  
{ 1.elb.amazonaws.com" }, "HealthCheckId": "hc-1234567890", "Failover": "PRIMARY" } } ] }
```

זמינות: Failover אוטומטי בין אזורים עם health checks מתקדמים.

עקרונות מפתח ליישום מוצלח | Key Principles for Successful Implementation

Security Best Practices

IAM Least Privilege Principle

עיקרון: מתן הרשאות מינימליות הנדרשות לביצוע משימה ספציפית.

```
Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": [ "ecs:DescribeServices", "ecs:UpdateService" ], "Resource": "arn:aws:ecs:us-east-1:123456789012:service/minday-service", { "Condition": { "StringEquals": { "aws:RequestTag/Environment": "production" } } } ]
```

יתרון: הפחתת סיכוני אבטחה ב-90% ומניעת הרשאות מיותרות.

Scalability Best Practices

Horizontal vs Vertical Scaling

אסטרטגיה: עדיפות לסקיילינג אופקי (הוספת instances) על פני אנכי (הגדלת resources).

- Horizontal Scaling:** הוספת ECS tasks אוטומטית
- Load Distribution:** ALB עם health checks
- Database Scaling:** MongoDB Atlas auto-scaling
- Cache Scaling:** ElastiCache cluster expansion

Monitoring Best Practices

Comprehensive Observability

גישה: ניטור מלא של כל שכבות המערכת עם correlation בין metrics.

- Infrastructure Metrics:** CPU, Memory, Network, Disk
- Application Metrics:** Response time, Error rates, Throughput
- Business Metrics:** User sessions, Transactions, Revenue
- Security Metrics:** Failed logins, WAF blocks, IAM events

המלצות וסיכום | Recommendations & Summary

תכנית פעולה מפורטת ליישום מוצלח | Detailed Action Plan for Successful Implementation

שלב 1: הכנה ותכנון (שבועות 1-2)

פעילויות קריטיות | Critical Activities

- Architecture Review**: סקירה מעמיקה של הארכיטקטורה הנוכחית
- Capacity Planning**: תכנון צרכים עתידיים ל-12-18 חודשים
- Security Assessment**: הערכת סיכונים אבטחה וצרכי compliance
- Team Training**: הכשרת הצוות על AWS services
- POC Development**: פיתוח proof-of-concept לאבטחת היתכנות

שלב 2: פיתוח תשתית (שבועות 3-6)

Infrastructure as Code | תשתית כקוד

- Terraform/CloudFormation**: הגדרת כל התשתית כקוד
- VPC Design**: תכנון רשת עם subnets נפרדים
- Security Groups**: הגדרת firewall rules מחמירים
- Monitoring Setup**: הגדרת CloudWatch dashboards
- Backup Configuration**: הגדרת backup strategies

שלב 3: מיגרציה הדרגתית (שבועות 7-10)

Blue-Green Deployment | פריסה כחול-ירוק

- Parallel Environment**: הפעלת המערכת החדשה במקביל
- Data Migration**: העברת נתונים הדרגתית
- Traffic Routing**: הפניית תעבורה הדרגתית
- Performance Testing**: בדיקות ביצועים מקיפות
- Rollback Plan**: תכנית חזרה למצב הקודם

סיכום מסחרי | Business Summary

ROI צפוי: החזר השקעה תוך 12-18 חודשים עם שיפור משמעותי בביצועים, זמינות ואבטחה.

Competitive Advantage: יכולת להתמודד עם צמיחה מהירה ותחרות גלובלית.

Future-Proof: תשתית מוכנה לעתיד עם AI, ML ו-technologies מתקדמות.

סיכום כללי | General Summary

המעבר ל-AWS יפתור את כל הבעיות הקיימות במערכת הנוכחית:

סקיילביליות: יכולת להתמודד עם 10x יותר משתמשים

ניתוח סיכונים מקיף ואסטרטגיות מיתון | Comprehensive Risk Analysis & Mitigation Strategies

סיכוני אבטחה | Security Risks

Data Breach Risk

סיכון: חשיפת נתונים רגישים עקב התקפות או טעויות אנוש.

```
Encryption at Rest & Transit aws s3api put-bucket-encryption \ --bucket minday-data \ --server-side- #  
encryption-configuration '{ "Rules": [{ "ApplyServerSideEncryptionByDefault": { "SSEAlgorithm": "AES256"  
' } ] } }
```

מיתון: הצפנה בזמן מנוחה ובמעבר, IAM roles מחמירים, WAF rules מתקדמים.

סיכוני זמינות | Availability Risks

Service Downtime Risk

סיכון: השבתת שירות עקב כשל בתשתית או באפליקציה.

```
Multi-AZ Deployment resource "aws_ecs_service" "minday_service" { name = "minday-service" cluster = #  
aws_ecs_cluster.minday_cluster.id desired_count = 2 deployment_maximum_percent = 200  
deployment_minimum_healthy_percent = 50 network_configuration { subnets = var.private_subnets  
security_groups = [aws_security_group.ecs_sg.id] } }
```

מיתון: Multi-AZ deployment, Auto-scaling, Health checks, Failover אוטומטי.

סיכוני ביצועים | Performance Risks

Performance Degradation Risk

סיכון: ירידה בביצועים עקב עומס גבוה או כשל בתשתית.

```
Auto Scaling Configuration { "AutoScalingGroupName": "minday-ecs-asg", "MinSize": 2, "MaxSize": 10, #  
"DesiredCapacity": 2, "TargetTrackingScalingPolicies": [{ "PolicyName": "CPUUtilization", "TargetValue":  
70.0 } ] }
```

מיתון: Auto-scaling, Load balancing, Caching, Performance monitoring.

אסטרטגיות מיתון כלליות | General Mitigation Strategies

- Monitoring & Alerting:** ניטור רציף עם התראות מיידיות
- Backup & Recovery:** תכניות backup ו-disaster recovery
- Security Best Practices:** יישום עקרונות אבטחה מתקדמים
- Performance Testing:** בדיקות ביצועים מקיפות
- Documentation:** תיעוד מפורט של כל התהליכים

סיכוני עלויות ואופטימיזציה | Cost Risks & Optimization

ניתוח סיכוני עלויות ואסטרטגיות אופטימיזציה | Cost Risk Analysis & Optimization Strategies

סיכוני עלויות | Cost Risks

Cost Overrun Risk

סיכון: עלויות גבוהות מהצפוי עקב שימוש לא יעיל במשאבים.

```
Cost Optimization Strategies aws ce get-cost-and-usage \ --time-period Start=2025-01-01,End=2025-01-31 # \ --granularity MONTHLY \ --metrics BlendedCost \ --group-by Type=DIMENSION,Key=SERVICE
```

מיתון: Cost monitoring, Reserved instances, Spot instances, Resource optimization.

אסטרטגיות אופטימיזציה | Optimization Strategies

Reserved Instances Strategy

אסטרטגיה: רכישת Reserved Instances לחיסכון של עד 75% בעלויות.

```
Reserved Instance Purchase aws ec2 describe-reserved-instances-offerings \ --instance-type t3.medium \ # --offering-type "All Upfront" \ --product-description "Linux/UNIX" \ --duration 31536000
```

חיסכון: עד 75% בעלויות compute עבור workloads יציבים.

Spot Instances Strategy

Cost-Effective Computing

אסטרטגיה: שימוש ב-Spot Instances עבור workloads גמישים.

```
Spot Fleet Configuration { "SpotFleetRequestConfig": { "TargetCapacity": 10, "IamFleetRole": # "arn:aws:iam::123456789012:role/spot-fleet-role", "LaunchSpecifications": [{ "ImageId": "ami-12345678", "InstanceType": "t3.medium", "SubnetId": "subnet-12345678", "WeightedCapacity": 1.0 } ] } }
```

חיסכון: עד 90% בעלויות compute עבור workloads שאינם קריטיים.

Auto Scaling Optimization

Dynamic Resource Management

אסטרטגיה: Auto-scaling חכם עם predictive scaling.

```
Predictive Scaling Policy { "AutoScalingGroupName": "minday-ecs-asg", "PolicyName": "predictive- # scaling-policy", "PolicyType": "PredictiveScaling", "PredictiveScalingConfiguration": { "MetricSpecification": { "TargetValue": 70.0, "PredefinedMetricPairSpecification": { "PredefinedMetricType": "ECSServiceAverageCPUUtilization" } }, "Mode": "ForecastAndScale" } }
```

יתרון: הוספת instances לפני עומס צפוי, חיסכון בזמן תגובה.