# PCI DSS 4.0 在 AWS 上的合規要求與測試工具

Version 1.2

## 1. 向客戶索取的資訊

| 所需資訊 | 目的 | 對應 PCI DSS 要求 |
| --- | --- | --- |
| 網路架構圖 | 驗證網路區隔與安全控制措施 | 1.1.1, 1.2.1, 1.3.1 |
| AWS 帳號存取權限 ( 唯讀 ) | 審查安全性設定與控制措施 | 多項要求 |
| 納入評估範圍的系統／元件清單 | 確認評估邊界 | 12.5.1, 12.5.2 |
| 持卡人資料環境 (CDE) 系統清單 | 確認所有元件皆已受到保護 | 2.1.1, 11.2.1 |
| DNS 架構與紀錄 | 驗證 DNS 的安全性設定 | 1.3.1, 2.2.5, 4.1.1 |

## 2. 使用開源工具的測試程序

### 網路安全控制 ( 對應 PCI DSS 需求 1.x )

| 測試項目 | 主要工具 | 開源替代工具 | 目的 |
| --- | --- | --- | --- |
| 安全群組規則分析 | aws ec2 describe-security-groups | Prowler, ScoutSuite | 驗證安全群組規則是否有效保護 CDE (1.2.1-1.2.8) |
| 網路區隔驗證 | aws ec2 describe-subnets, aws ec2 describe-route-tables | Nmap, Zmap | 確認 CDE 與不可信網路之間的隔離 (1.3.1-1.3.2) |
| 公網資源的安全控制 | aws ec2 describe-instances, aws elb describe-load-balancers | Nmap, Nuclei | 驗證公開資源的防護是否到位 (1.5.1) |
| VPC 端點控制 | aws ec2 describe-vpc-endpoints | ScoutSuite | 驗證服務端點的存取控制 (1.3.1-1.3.2) |
| 網路 ACL 審查 | aws ec2 describe-network-acls | Prowler, CloudSploit | 驗證網路存取控制清單是否合理 (1.2.1-1.2.8) |

#### 網路安全控制範例指令

| 測試項目 | 範例指令 |
| --- | --- |
| 安全群組規則分析 | aws ec2 describe-security-groups --filters Name=group-name,Values=\*cde\* --query "SecurityGroups[\*].{Name:GroupName,ID:GroupId,IngressRules:IpPermissions[\*]}" --output json |
| 網路區隔驗證 | aws ec2 describe-subnets --filters Name=tag:Environment,Values=PCI --query "Subnets[\*].{ID:SubnetId,CIDR:CidrBlock,VPC:VpcId,AZ:AvailabilityZone}" --output table |
| 公網資源的安全控制 | aws ec2 describe-instances --filters Name=instance-state-name,Values=running --query "Reservations[\*].Instances[\*].{ID:InstanceId,PublicIP:PublicIpAddress,PrivateIP:PrivateIpAddress,Type:InstanceType,SecurityGroups:SecurityGroups[\*].GroupId}" --output table |
| VPC 端點控制 | aws ec2 describe-vpc-endpoints --query "VpcEndpoints[\*].{ID:VpcEndpointId,Service:ServiceName,VPC:VpcId,State:State,Policy:PolicyDocument}" --output json |
| 網路 ACL 審查 | aws ec2 describe-network-acls --query "NetworkAcls[\*].{ID:NetworkAclId,VPC:VpcId,Inbound:Entries[?Egress==\false`]}” –output json` |

### 系統組態標準（對應需求 2.x）

| 測試項目 | 主要工具 | 開源替代工具 | 目的 |
| --- | --- | --- | --- |
| 系統強化評估 | aws ec2 describe-instances, aws ssm describe-instance-information | Lynis, OpenSCAP | 驗證系統是否已安全設定 (2.2.1-2.2.7) |
| 管理存取審查 | aws iam list-roles, aws iam get-policy | CloudSploit, InSpec | 確認管理存取已加密 (2.3.1-2.3.2) |
| 預設設定分析 | aws rds describe-db-instances | AWS Config Rules, CloudSploit | 辨識預設帳密與設定 (2.1.1) |
| EC2 組態分析 | aws ec2 describe-instances | InSpec, Prowler | 驗證系統元件是否已正確設定 (2.2.1-2.2.5) |
| AMI 映像檔審查 | aws ec2 describe-images | Amazon Inspector | 驗證基礎映像檔是否安全 (2.2.1-2.2.5) |

#### 系統組態標準範例指令

| 測試項目 | 範例指令 |
| --- | --- |
| 系統強化評估 | aws ssm describe-instance-information --query "InstanceInformationList[\*].{InstanceId:InstanceId,PlatformName:PlatformName,PlatformVersion:PlatformVersion,PingStatus:PingStatus}" --output table |
| 管理存取審查 | aws iam list-roles --query "Roles[?contains(RoleName, 'admin')].{RoleName:RoleName,CreateDate:CreateDate,Path:Path}" --output table |
| 預設設定分析 | aws rds describe-db-instances --query "DBInstances[\*].{DBIdentifier:DBInstanceIdentifier,Engine:Engine,MasterUsername:MasterUsername,MultiAZ:MultiAZ,Encrypted:StorageEncrypted}" --output table |
| EC2 組態分析 | aws ec2 describe-instances --query "Reservations[\*].Instances[\*].{ID:InstanceId,AMI:ImageId,Type:InstanceType,State:State.Name,LaunchTime:LaunchTime}" --output table |
| AMI 映像檔審查 | aws ec2 describe-images --owners self --query "Images[\*].{ImageId:ImageId,Name:Name,CreationDate:CreationDate,Public:Public}" --output table |

### 資料保護機制（需求 3.x, 4.x）

| 測試項目 | 主要工具 | 開源替代工具 | 目的 |
| --- | --- | --- | --- |
| 加密設定分析 | aws kms list-keys, aws s3 get-bucket-encryption | Prowler, CloudSploit | 驗證存儲中的持卡人資料 (CHD) 是否已加密(3.4.1–3.5.1) |
| 密鑰管理審查 | aws kms describe-key | ScoutSuite, CloudMapper | 評估加密密鑰的安全性(3.6.1–3.7.1) |
| 憑證驗證 | aws acm list-certificates | SSLyze, testssl.sh | 確認憑證的正確實作(4.1.1) |
| S3 加密設定 | aws s3 get-bucket-encryption | S3Scanner, CloudSploit | 驗證 S3 的加密設定(3.4.1) |
| EBS 卷加密設定 | aws ec2 describe-volumes | Prowler | 驗證 EBS 卷的加密設定 (3.4.1) |

#### 資料保護機制範例指令

| 測試項目 | 範例指令 |
| --- | --- |
| 加密設定分析 | aws kms list-keys --query "Keys[\*].KeyId" --output text \| xargs -I {} aws kms describe-key --key-id {} --query "KeyMetadata.{KeyId:KeyId,Description:Description,Enabled:Enabled,KeyUsage:KeyUsage}" --output table |
| 密鑰管理審查 | aws kms list-aliases --query "Aliases[?contains(AliasName, 'pci') \|\| contains(AliasName, 'payment')].TargetKeyId" --output text \| xargs -I {} aws kms get-key-rotation-status --key-id {} |
| 憑證驗證 | aws acm list-certificates --query "CertificateSummaryList[\*].{CertificateArn:CertificateArn,DomainName:DomainName,Status:Status}" --output table |
| S3 加密設定 | aws s3api list-buckets --query "Buckets[\*].Name" --output text \| xargs -I {} aws s3api get-bucket-encryption --bucket {} 2>/dev/null \|\| echo "Bucket {} not encrypted" |
| EBS 卷加密設定 | aws ec2 describe-volumes --query "Volumes[\*].{VolumeId:VolumeId,State:State,Encrypted:Encrypted,Size:Size,AZ:AvailabilityZone}" --output table |

### 存取控制評估（需求 7.x, 8.x）

| 測試項目 | 主要工具 | 開源替代工具 | 目的 |
| --- | --- | --- | --- |
| IAM 政策分析 | aws iam get-account-authorization-details | IAM Access Analyzer, Prowler | 驗證最小權限原則(7.1.1-7.3.3) |
| 存取路徑映射 | aws iam list-roles, aws iam list-users | CloudMapper, PMapper | 辨識通往持卡人資料系統的存取路徑(7.2.4) |
| 認證機制 | aws iam list-virtual-mfa-devices | Prowler, ScoutSuite | 審查 MFA 和認證控制(8.3.1-8.6.3) |
| 密碼政策審查 | aws iam get-account-password-policy | CloudSploit, Prowler | 驗證密碼政策合規性(8.6.1-8.6.3) |
| 跨帳戶存取審查 | aws organizations list-accounts | CloudMapper, Prowler | 審查跨帳戶存取(7.3.1-7.3.3) |

#### 存取控制評估範例指令

| 測試項目 | 範例指令 |
| --- | --- |
| IAM 政策分析 | aws iam get-account-authorization-details --filter IAM-USER --query "UserDetailList[\*].{UserName:UserName,AttachedManagedPolicies:AttachedManagedPolicies[\*].PolicyName}" --output json |
| 存取路徑映射 | aws iam list-users --query "Users[\*].{UserName:UserName,UserId:UserId,CreateDate:CreateDate}" --output table && aws iam list-roles --query "Roles[\*].{RoleName:RoleName,CreateDate:CreateDate}" --output table |
| 認證機制 | aws iam list-virtual-mfa-devices --query "VirtualMFADevices[\*].{User:User.UserName,SerialNumber:SerialNumber,EnableDate:EnableDate}" --output table |
| 密碼政策審查 | aws iam get-account-password-policy --query "PasswordPolicy.{MinimumPasswordLength:MinimumPasswordLength,RequireSymbols:RequireSymbols,RequireNumbers:RequireNumbers,RequireUppercaseCharacters:RequireUppercaseCharacters,RequireLowercaseCharacters:RequireLowercaseCharacters,PasswordReusePrevention:PasswordReusePrevention,MaxPasswordAge:MaxPasswordAge}" --output table |
| 跨帳戶存取審查 | aws organizations list-accounts --query "Accounts[\*].{Id:Id,Name:Name,Status:Status,Email:Email}" --output table |

### 漏洞管理（需求 6.x, 11.x）

| 測試項目 | 主要工具 | 開源替代工具 | 目的 |
| --- | --- | --- | --- |
| WAF 配置審查 | aws waf list-web-acls | OWASP ZAP, Nuclei | 驗證網路應用防火牆 (WAF) 保護措施 (6.4.1-6.4.2) |
| 系統元件分析 | aws ecr describe-images | Trivy, Clair | 辨識過時的軟體元件(6.3.1-6.3.3) |
| 補丁管理審查 | aws ssm describe-patch-baselines | OpenVAS, Prowler | 驗證補丁狀態與流程(6.3.3) |
| 漏洞掃描 | aws inspector list-findings | OWASP ZAP, Nikto | 辨識系統中的漏洞(11.3.1-11.3.2) |
| GuardDuty 發現 | aws guardduty list-findings | Prowler, CloudSploit | 審查安全發現(11.4.1-11.4.5) |

#### 漏洞管理範例指令

| 測試項目 | 範例指令 |
| --- | --- |
| WAF 配置審查 | aws wafv2 list-web-acls --scope REGIONAL --query "WebACLs[\*].{Name:Name,Id:Id,ARN:ARN}" --output table |
| 系統元件分析 | aws ecr describe-repositories --query "repositories[\*].{Name:repositoryName,URI:repositoryUri,Created:createdAt}" --output table |
| 補丁管理審查 | aws ssm describe-patch-baselines --query "BaselineIdentities[\*].{Name:BaselineName,ID:BaselineId,OperatingSystem:OperatingSystem,Description:BaselineDescription}" --output table |
| 漏洞掃描 | aws inspector2 list-findings --filter "findingStatus={comparison=EQUALS,value=ACTIVE}" --max-results 10 --query "findings[\*].{Title:title,Severity:severity,ResourceId:resources[0].id}" --output table |
| GuardDuty 發現 | aws guardduty list-detectors --query "DetectorIds" --output text \| xargs -I {} aws guardduty list-findings --detector-id {} --finding-criteria '{"Criterion":{"severity":{"Eq":["8","9"]}}}' |

### 日誌與監控（需求 10.x, 11.x）

| 測試項目 | 主要工具 | 開源替代工具 | 目的 |
| --- | --- | --- | --- |
| CloudTrail 配置 | aws cloudtrail describe-trails | CloudSploit, Prowler | 驗證審計日誌是否已啟用 (10.2.1-10.2.2) |
| 日誌配置審查 | aws logs describe-log-groups | ELK Stack, Graylog | 確認正確的日誌設定與保留期 (10.5.1-10.7.3) |
| CloudWatch 警報審查 | aws cloudwatch describe-alarms | Grafana, Prometheus | 驗證持續監控 (11.4.1-11.4.5) |
| 日誌內容驗證 | aws logs filter-log-events | Logstash, Loki | 確認記錄所需的事件(10.2.1-10.2.2) |
| VPC 流量日誌審查 | aws ec2 describe-flow-logs | Zeek, Suricata | 驗證網路流量日誌(10.2.2, 10.3.2) |

#### 日誌與監控範例指令

| 測試項目 | 範例指令 |
| --- | --- |
| CloudTrail 配置 | aws cloudtrail describe-trails --query "trailList[\*].{Name:Name,IsMultiRegion:IsMultiRegionTrail,LoggingEnabled:IsLogging,HomeRegion:HomeRegion}" --output table |
| 日誌配置審查 | aws logs describe-log-groups --query "logGroups[\*].{LogGroupName:logGroupName,RetentionInDays:retentionInDays,StoredBytes:storedBytes}" --output table |
| CloudWatch 警報審查 | aws cloudwatch describe-alarms --state-value ALARM --query "MetricAlarms[\*].{Name:AlarmName,Metric:MetricName,Namespace:Namespace,State:StateValue}" --output table |
| 日誌內容驗證 | aws logs filter-log-events --log-group-name /aws/cloudtrail --filter-pattern "{$.eventName = ConsoleLogin}" --limit 5 |
| VPC 流量日誌審查 | aws ec2 describe-flow-logs --query "FlowLogs[\*].{Id:FlowLogId,LogDestination:LogDestination,ResourceId:ResourceId,TrafficType:TrafficType,Status:FlowLogStatus}" --output table |

### DNS 安全性評估

| 測試項目 | 主要工具 | 開源替代工具 | 目的 |
| --- | --- | --- | --- |
| Route 53 配置分析 | aws route53 list-hosted-zones | DNSRecon, Fierce | 驗證 CDE 元件的正確 DNS 設定 (1.2.1, 2.2.5) |
| DNS 安全擴展 | aws route53 get-hosted-zone | DNSViz, delv | 確認 DNSSEC 實作 (4.1.1) |
| Zone 轉移測試 | aws route53 list-resource-record-sets | DNSenum, dig | 辨識區域轉移漏洞 (1.3.1, 11.3.1) |
| DNS 記錄列舉 | aws route53 list-resource-record-sets | Amass, Sublist3r | 繪製 DNS 基礎架構並驗證區隔 (1.3.1) |
| 網域安全性分析 | aws route53domains list-domains | dnstwist, DNSTwist | 偵測打字駭客與網域安全性問題 (6.4.1, 11.3.1) |

#### DNS 安全性評估範例指令

| 測試項目 | 範例指令 |
| --- | --- |
| Route 53 配置分析 | aws route53 list-hosted-zones --query "HostedZones[\*].{Name:Name,Id:Id,Private:Config.PrivateZone}" --output table |
| DNS 安全擴展 | aws route53 list-hosted-zones --query "HostedZones[\*].Id" --output text \| xargs -I {} aws route53 get-dnssec --hosted-zone-id {} 2>/dev/null \|\| echo "DNSSEC not enabled" |
| Zone 轉移測試 | aws route53 list-hosted-zones --query "HostedZones[\*].Id" --output text \| head -1 \| xargs -I {} aws route53 list-resource-record-sets --hosted-zone-id {} --query "ResourceRecordSets[?Type=='NS']" |
| DNS 記錄列舉 | aws route53 list-hosted-zones --query "HostedZones[\*].Id" --output text \| head -1 \| xargs -I {} aws route53 list-resource-record-sets --hosted-zone-id {} --query "ResourceRecordSets[\*].{Name:Name,Type:Type,TTL:TTL}" --output table |
| 網域安全性分析 | aws route53domains list-domains --query "Domains[\*].{DomainName:DomainName,Expiry:Expiry,AutoRenew:AutoRenew}" --output table |

## 3. 所需的 AWS 用戶帳戶以進行評估

對於 PCI DSS 4.0 評估，需要一個具有適當唯讀權限的普通 AWS IAM 用戶帳戶。此帳戶可用來正常通過 AWS 管理控制台和 AWS CLI 登入。

### 所需的用戶帳戶權限

應為用戶帳戶分配以下 AWS 管理的策略，以確保完整的評估能力：

| 策略 | 目的 | PCI DSS 要求 |
| --- | --- | --- |
| ReadOnlyAccess | 基本的唯讀存取大多數 AWS 資源 | Multiple |
| SecurityAudit | 存取與安全相關的資訊 | 7.1, 7.2, 8.1, 11.3 |
| AmazonInspector2ReadOnlyAccess | 存取 Inspector 漏洞發現 | 6.3, 11.3 |
| AmazonGuardDutyReadOnlyAccess | 存取 GuardDuty 安全發現 | 10.6, 11.4 |
| CloudWatchReadOnlyAccess | 存取監控與指標資料 | 10.4, 10.6, 11.5 |
| AWSCloudTrailReadOnlyAccess | 存取審計日誌資料 | 10.2, 10.3, 10.5 |
| AWSConfigUserAccess | 存取 AWS Config 合規性資料 | 1.2.7, 2.2.1, 6.3.3 |
| AmazonVPCReadOnlyAccess | 存取 VPC 配置 | 1.2, 1.3, 1.4 |
| IAMReadOnlyAccess | 存取 IAM 配置 | 7.1, 7.2, 8.1, 8.3 |
| AmazonS3ReadOnlyAccess | 存取 S3 配置 | 3.4, 9.4, 10.5 |
| AWSSecurityHubReadOnlyAccess | 存取 Security Hub 發現 | 6.3, 11.3, 11.4 |

### 自訂 PCI 評估者策略

建議創建一個自訂的單一策略，而非分配多個預定義的策略，這樣可以更精確地授權評估所需的權限。此方法遵循最小權限原則 ( 需求：7.2.1)，並確保能存取所有必要的資源進行完整的 PCI DSS 評估：

{  
 "Version": "2012-10-17",  
 "Statement": [  
 {  
 "Effect": "Allow",  
 "Action": [  
 "acm:List\*",  
 "acm:Describe\*",  
 "cloudtrail:Describe\*",  
 "cloudtrail:Get\*",  
 "cloudtrail:List\*",  
 "cloudtrail:LookupEvents",  
 "cloudwatch:Describe\*",  
 "cloudwatch:Get\*",  
 "cloudwatch:List\*",  
 "config:BatchGetResourceConfig",  
 "config:Get\*",  
 "config:List\*",  
 "config:SelectResourceConfig",  
 "ec2:Describe\*",  
 "ecr:Describe\*",  
 "ecr:List\*",  
 "ecs:Describe\*",  
 "ecs:List\*",  
 "eks:Describe\*",  
 "eks:List\*",  
 "elasticloadbalancing:Describe\*",  
 "events:Describe\*",  
 "events:List\*",  
 "guardduty:Get\*",  
 "guardduty:List\*",  
 "iam:GenerateCredentialReport",  
 "iam:GenerateServiceLastAccessedDetails",  
 "iam:Get\*",  
 "iam:List\*",  
 "iam:SimulatePrincipalPolicy",  
 "inspector2:List\*",  
 "inspector2:Describe\*",  
 "inspector2:Get\*",  
 "kms:Describe\*",  
 "kms:Get\*",  
 "kms:List\*",  
 "lambda:GetFunction",  
 "lambda:GetFunctionConfiguration",  
 "lambda:List\*",  
 "logs:Describe\*",  
 "logs:FilterLogEvents",  
 "logs:Get\*",  
 "logs:StartQuery",  
 "logs:StopQuery",  
 "logs:TestMetricFilter",  
 "macie2:Describe\*",  
 "macie2:Get\*",  
 "macie2:List\*",  
 "organizations:Describe\*",  
 "organizations:List\*",  
 "rds:Describe\*",  
 "route53:Get\*",  
 "route53:List\*",  
 "route53domains:GetDomainDetail",  
 "route53domains:List\*",  
 "s3:Get\*",  
 "s3:List\*",  
 "secretsmanager:List\*",  
 "secretsmanager:Describe\*",  
 "securityhub:BatchGetSecurityControls",  
 "securityhub:Get\*",  
 "securityhub:List\*",  
 "ssm:Describe\*",  
 "ssm:Get\*",  
 "ssm:List\*",  
 "shield:Describe\*",  
 "shield:Get\*",  
 "shield:List\*",  
 "sns:Get\*",  
 "sns:List\*",  
 "waf:Get\*",  
 "waf:List\*",  
 "wafv2:Get\*",  
 "wafv2:List\*"  
 ],  
 "Resource": "\*"  
 }  
 ]  
}

### 建立與指派自訂策略

# 創建本地 JSON 文件，並將策略儲存於檔案中  
cat > pci-assessor-policy.json << 'EOF'  
{  
 "Version": "2012-10-17",  
 "Statement": [  
 {  
 "Effect": "Allow",  
 "Action": [  
 "acm:List\*",  
 "acm:Describe\*",  
 "cloudtrail:Describe\*",  
 "cloudtrail:Get\*",  
 "cloudtrail:List\*",  
 "cloudwatch:Describe\*",  
 "cloudwatch:Get\*",  
 "cloudwatch:List\*",  
 "config:Get\*",  
 "config:List\*",  
 "ec2:Describe\*",  
 "ecr:Describe\*",  
 "ecr:List\*",  
 "ecs:Describe\*",  
 "ecs:List\*",  
 "eks:Describe\*",  
 "eks:List\*",  
 "elasticloadbalancing:Describe\*",  
 "guardduty:Get\*",  
 "guardduty:List\*",  
 "iam:Get\*",  
 "iam:List\*",  
 "inspector2:List\*",  
 "inspector2:Describe\*",  
 "kms:Describe\*",  
 "kms:Get\*",  
 "kms:List\*",  
 "lambda:List\*",  
 "logs:Describe\*",  
 "logs:FilterLogEvents",  
 "logs:Get\*",  
 "organizations:List\*",  
 "rds:Describe\*",  
 "route53:Get\*",  
 "route53:List\*",  
 "route53domains:List\*",  
 "s3:Get\*",  
 "s3:List\*",  
 "secretsmanager:List\*",  
 "secretsmanager:Describe\*",  
 "securityhub:Get\*",  
 "securityhub:List\*",  
 "ssm:Describe\*",  
 "ssm:Get\*",  
 "ssm:List\*",  
 "waf:Get\*",  
 "waf:List\*",  
 "wafv2:Get\*",  
 "wafv2:List\*"  
 ],  
 "Resource": "\*"  
 }  
 ]  
}  
EOF  
  
# 創建自訂策略  
aws iam create-policy \  
 --policy-name PCI\_DSS\_Assessor \  
 --policy-document file://pci-assessor-policy.json  
  
# 創建用於評估的用戶  
aws iam create-user --user-name pci\_assessor  
  
# 將策略指派給該用戶  
aws iam attach-user-policy \  
 --user-name pci\_assessor \  
 --policy-arn arn:aws:iam::ACCOUNT\_ID:policy/PCI\_DSS\_Assessor

### 帳戶設置驗證

在策略分配後，通過測試以下關鍵服務的存取來驗證用戶是否擁有正確的存取權限，這些服務是進行 PCI DSS 評估所需的：

# 創建評估者帳戶的訪問金鑰  
aws iam create-access-key --user-name pci\_assessor  
  
# 使用新憑證配置 AWS CLI  
aws configure  
  
# 驗證網路安全控制的存取（需求 1.x）  
aws ec2 describe-security-groups --region us-east-1 --max-items 1  
aws ec2 describe-network-acls --region us-east-1 --max-items 1  
aws ec2 describe-subnets --region us-east-1 --max-items 1  
  
# 驗證系統配置資料的存取（需求 2.x）  
aws ec2 describe-instances --region us-east-1 --max-items 1  
aws rds describe-db-instances --region us-east-1 --max-items 1  
  
# 驗證加密資料的存取（需求 3.x）  
aws kms list-keys --region us-east-1 --max-items 1  
aws s3 get-bucket-encryption --bucket <sample-bucket-name> 2>/dev/null || echo "Access verified"  
  
# 驗證存取控制資料的存取（需求 7.x, 8.x）  
aws iam list-users --max-items 1  
aws iam get-account-password-policy 2>/dev/null || echo "Access verified"  
  
# 驗證日誌與監控資料的存取（需求 10.x）  
aws cloudtrail describe-trails --region us-east-1  
aws logs describe-log-groups --region us-east-1 --max-items 1  
aws cloudwatch describe-alarms --region us-east-1 --max-items 1  
  
# 驗證漏洞管理資料的存取（需求 6.x, 11.x）  
aws inspector2 list-findings --region us-east-1 --max-items 1 2>/dev/null || echo "Access verified"  
aws securityhub list-findings --region us-east-1 --max-items 1 2>/dev/null || echo "Access verified"  
aws guardduty list-findings --region us-east-1 --max-items 1 2>/dev/null || echo "Access verified"  
  
# 列出已附加到用戶的策略以確認設置正確  
aws iam list-attached-user-policies --user-name pci\_assessor

這樣的全面測試方法可確保評估帳戶擁有正確的唯讀存取權限，並能夠對所有需要的系統進行完整的 PCI DSS 4.0 合規性驗證，涵蓋每個主要需求類別。

### 附註

1. 所有測試將在一台運行 Ubuntu 24.04 的虛擬機上執行，硬體規格如下：
   * 8GB RAM
   * 2 CPUs
   * 50GB SSD
2. 不會在任何 AWS 系統中安裝工具。
3. 所有測試皆為非侵入性，並使用具有唯讀權限的 AWS 帳戶執行。
4. 主要使用 AWS CLI 工具進行合規性驗證。
5. 所有提到的工具皆為開源且免費使用。
6. 此評估流程僅用於 PCI DSS 4.0 合規性驗證，非滲透測試。
7. AWS CLI 指令為主要的驗證手段，開源工具則作為額外驗證補充。
8. IAM 權限配置遵循 PCI DSS 4.0 第 7.2.1 條：最小權限原則。
9. AWS Config 規則與 AWS Security Hub 可用於執行額外自動化的 PCI DSS 合規性檢查。
10. 根據第 11.3.2 項，仍需由通過認證的 ASV 掃描供應商執行對外漏洞掃描。
11. AWS 原生服務如 Amazon Inspector、GuardDuty 和 Security Hub 可協助滿足 第 6.3.1、10.7.2、11.4.1 條要求，但不能取代所有控制措施。