```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Description": "This template creates necessary resources to grant access to Cloud Conformity",
  "Parameters": {
    "AccountId": {
      "Type": "String",
      "Description": "Cloud Conformity AWS Account ID"
    },
    "ExternalId": {
      "Type": "String",
      "Description": "The primary function of the External ID is to address and prevent the 'confused deputy' problem"
    }
  },
  "Resources": {
    "CloudConformityCustomPolicyPart1": {
      "Type": "AWS::IAM::ManagedPolicy",
      "Properties": {
        "ManagedPolicyName": "CloudConformityPart1",
        "Description": "Cloud Conformity Custom Policy Part 1",
        "PolicyDocument": {
          "Version": "2012-10-17",
          "Statement": [
            {
              "Effect": "Allow",
              "Action": [
                "acm:DescribeCertificate",
                "acm:ListCertificates",
                "acm:ListTagsForCertificate",
                "apigateway:GET",
                "autoscaling:DescribeAccountLimits",
```

"autoscaling:DescribeAutoScalingGroups",

"autoscaling:DescribeAutoScalingInstances",

"autoscaling:DescribeLaunchConfigurations",

"autoscaling:DescribeLoadBalancerTargetGroups",

"autoscaling:DescribeLoadBalancers",

"autoscaling:DescribeNotificationConfigurations",

"autoscaling:DescribeTags",

"cloudformation:DescribeAccountLimits",

"cloudformation:DescribeStackDriftDetectionStatus",

"cloudformation:DescribeStacks",

"cloudformation:DetectStackDrift",

"cloudformation:GetStackPolicy",

"cloudformation:ListStacks",

"cloudfront:GetDistribution",

"cloudfront:ListDistributions",

"cloudfront:ListTagsForResource",

"cloudtrail:DescribeTrails",

"cloudtrail:GetTrailStatus",

"cloudtrail:GetEventSelectors",

"cloudtrail:ListTags",

"cloudwatch:DescribeAlarms",

"cloudwatch:DescribeAlarmsForMetric",

"cloudwatch:GetMetricStatistics",

"cloudwatch:ListMetrics",

"config:DescribeComplianceByConfigRule",

"config:DescribeConfigRules",

"config:DescribeConfigurationRecorderStatus",

"config:DescribeConfigurationRecorders",

"config:DescribeDeliveryChannelStatus",

"config:DescribeDeliveryChannels",

"config:GetComplianceDetailsByConfigRule",

"config:GetResourceConfigHistory",

"config:SelectResourceConfig",

"dynamodb:DescribeContinuousBackups",

"dynamodb:DescribeLimits",

"dynamodb:DescribeTable",

"dynamodb:ListBackups",

"dynamodb:ListTables",

"dynamodb:ListTagsOfResource",

"ec2:DescribeAccountAttributes",

"ec2:DescribeAddresses",

"ec2:DescribeEgressOnlyInternetGateways",

"ec2:DescribeFlowLogs",

"ec2:DescribeImages",

"ec2:DescribeInstanceAttribute",

"ec2:DescribeInstanceStatus",

"ec2:DescribeInstances",

"ec2:DescribeInternetGateways",

"ec2:DescribeKeyPairs",

"ec2:DescribeNatGateways",

"ec2:DescribeNetworkAcls",

"ec2:DescribeNetworkInterfaces",

"ec2:DescribeReservedInstances",

"ec2:DescribeRouteTables",

"ec2:DescribeSecurityGroupReferences",

"ec2:DescribeSecurityGroups",

"ec2:DescribeSnapshots",

"ec2:DescribeSnapshotAttribute",

"ec2:DescribeSubnets",

"ec2:DescribeTags",

"ec2:DescribeVolumes",

"ec2:DescribeVpcAttribute",

"ec2:DescribeVpcEndpoints",

"ec2:DescribeVpcPeeringConnections",

"ec2:DescribeVpcs",

"ec2:DescribeVpnConnections",

"ec2:DescribeVpnGateways",

"ec2:GetEbsEncryptionByDefault",

"elasticfilesystem:DescribeFileSystems",

"elasticfilesystem:DescribeTags",

"elasticmapreduce:DescribeCluster",

"elasticmapreduce:ListClusters",

"elasticmapreduce:ListInstances",

"es:DescribeElasticsearchDomain",

"es:DescribeElasticsearchDomainConfig",

"es:DescribeElasticsearchDomains",

"es:DescribeElasticsearchInstanceTypeLimits",

"es:DescribeReservedElasticsearchInstanceOfferings",

"es:DescribeReservedElasticsearchInstances",

"es:ListDomainNames",

"es:ListElasticsearchInstanceTypes",

"es:ListElasticsearchVersions",

"es:ListTags",

"elasticache:DescribeCacheClusters",

"elasticache:DescribeReplicationGroups",

"elasticache:DescribeReservedCacheNodes",

"elasticache:ListTagsForResource",

"elasticloadbalancing:DescribeListeners",

"elasticloadbalancing:DescribeLoadBalancerAttributes",

"elasticloadbalancing:DescribeLoadBalancerPolicies",

"elasticloadbalancing:DescribeLoadBalancers",

"elasticloadbalancing:DescribeTags",

"elasticloadbalancing:DescribeTargetGroups",

"elasticloadbalancing:DescribeTargetHealth",

"elasticloadbalancing:DescribeRules",

"iam:GenerateCredentialReport",

"iam:GetAccessKeyLastUsed",

"iam:GetAccountAuthorizationDetails",

"iam:GetAccountPasswordPolicy",

"iam:GetAccountSummary",

"iam:GetCredentialReport",

"iam:GetGroup",

"iam:GetGroupPolicy",

"iam:GetLoginProfile",

"iam:GetOpenIDConnectProvider",

"iam:GetPolicy",

"iam:GetPolicyVersion",

"iam:GetRole",

"iam:GetRolePolicy",

"iam:GetSAMLProvider",

"iam:GetServerCertificate",

"iam:GetUser",

"iam:GetUserPolicy",

"iam:ListAccessKeys",

"iam:ListAccountAliases",

"iam:ListAttachedGroupPolicies",

"iam:ListAttachedRolePolicies",

"iam:ListAttachedUserPolicies",

"iam:ListEntitiesForPolicy",

"iam:ListGroupPolicies",

"iam:ListGroups",

"iam:ListInstanceProfiles",

"iam:ListInstanceProfilesForRole",

"iam:ListMFADevices",

"iam:ListOpenIDConnectProviders",

"iam:ListPolicies",

"iam:ListPolicyTags",

"iam:ListPolicyVersions",

"iam:ListRolePolicies",

"iam:ListRoleTags",

"iam:ListRoles",

"iam:ListSAMLProviders",

"iam:ListSSHPublicKeys",

"iam:ListServerCertificates",

"iam:ListUserPolicies",

"iam:ListUserTags",

"iam:ListUsers",

"iam:ListVirtualMFADevices",

"kms:DescribeKey",

"kms:GetKeyPolicy",

"kms:GetKeyRotationStatus",

"kms:ListAliases",

"kms:ListGrants",

"kms:ListKeyPolicies",

"kms:ListKeys",

"kms:ListResourceTags",

"lambda:GetAccountSettings",

"lambda:GetFunctionConfiguration",

"lambda:GetPolicy",

"lambda:ListEventSourceMappings",

"lambda:ListFunctions",

"lambda:ListTags",

"lambda:ListFunctionUrlConfigs",

"lambda:ListLayers",

"logs:DescribeLogGroups",

"logs:DescribeMetricFilters",

"rds:DescribeAccountAttributes",

"rds:DescribeDBClusters",

"rds:DescribeDBInstances",

"rds:DescribeDBSecurityGroups",

"rds:DescribeDBSnapshotAttributes",

"rds:DescribeDBSnapshots",

"rds:DescribeDBParameters",

"rds:DescribeDBParameterGroups",

"rds:DescribeEvents",

"rds:DescribeEventSubscriptions",

"rds:DescribeReservedDBInstances",

"rds:ListTagsForResource",

"redshift:DescribeClusterParameterGroups",

"redshift:DescribeClusterParameters",

"redshift:DescribeClusters",

"redshift:DescribeLoggingStatus",

"redshift:DescribeReservedNodes",

"redshift:DescribeTags",

"route53:GetDNSSEC",

"route53:GetGeoLocation",

"route53:ListHostedZones",

"route53:ListResourceRecordSets",

"route53:ListTagsForResource",

"route53domains:ListDomains",

"route53domains:ListTagsForDomain",

"ses:GetIdentityDkimAttributes",

"ses:GetIdentityPolicies",

"ses:GetIdentityVerificationAttributes",

"ses:ListIdentities",

"ses:ListIdentityPolicies",

```
            "sns:GetTopicAttributes",
            "sns:ListTopics",
            "sns:ListSubscriptionsByTopic",
            "sns:ListTagsForResource",
            "sqs:GetQueueAttributes",
            "sqs:ListQueues",
            "sqs:ListQueueTags",
            "tag:GetResources",
            "tag:GetTagKeys",
            "tag:GetTagValues"
          ],
          "Resource": [
            "*"
          ]
        }
      ]
    }
  }
},
"CloudConformityCustomPolicyPart2": {
  "Type": "AWS::IAM::ManagedPolicy",
  "Properties": {
    "ManagedPolicyName": "CloudConformityPart2",
    "Description": "Cloud Conformity Custom Policy Part 2",
    "PolicyDocument": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Action": [
            "access-analyzer:ListAnalyzers",
```

"access-analyzer:ListFindings",

"application-autoscaling:DescribeScalableTargets",

"application-autoscaling:DescribeScalingActivities",

"application-autoscaling:DescribeScalingPolicies",

"application-autoscaling:DescribeScheduledActions",

"athena:GetQueryExecution",

"athena:ListQueryExecutions",

"athena:ListTagsForResource",

"backup:DescribeBackupVault",

"backup:ListBackupVaults",

"backup:ListRecoveryPointsByResource",

"backup:GetBackupVaultAccessPolicy",

"ce:GetAnomalies",

"ce:GetAnomalyMonitors",

"cloudwatch:GetMetricData",

"dax:DescribeClusters",

"dax:ListTags",

"dms:DescribeReplicationInstances",

"dms:ListTagsForResource",

"ds:DescribeDirectories",

"ds:ListTagsForResource",

"ec2:DescribeTransitGatewayPeeringAttachments",

"ec2:SearchTransitGatewayRoutes",

"ec2:DescribeTransitGatewayRouteTables",

"ec2:DescribeTransitGateways",

"ec2:DescribeTransitGatewayAttachments",

"elasticbeanstalk:DescribeConfigurationSettings",

"elasticbeanstalk:DescribeEnvironments",

"ecr:DescribeRepositories",

"ecr:GetRepositoryPolicy",

"ecr:GetLifecyclePolicy",

"ecr:DescribeImages",

"eks:DescribeCluster",

"eks:ListClusters",

"events:DescribeEventBus",

"events:ListRules",

"firehose:DescribeDeliveryStream",

"firehose:ListDeliveryStreams",

"firehose:ListTagsForDeliveryStream",

"kafka:DescribeCluster",

"kafka:ListClusters",

"kafka:ListNodes",

"mq:DescribeBroker",

"mq:ListBrokers",

"glue:GetDataCatalogEncryptionSettings",

"glue:GetSecurityConfiguration",

"glue:GetSecurityConfigurations",

"glue:GetDatabases",

"guardduty:GetDetector",

"guardduty:GetFindings",

"guardduty:ListDetectors",

"guardduty:ListFindings",

"health:DescribeAffectedEntities",

"health:DescribeEventDetails",

"health:DescribeEvents",

"inspector:DescribeAssessmentRuns",

"inspector:DescribeAssessmentTargets",

"inspector:DescribeAssessmentTemplates",

"inspector:DescribeExclusions",

"inspector:DescribeFindings",

"inspector:DescribeResourceGroups",

"inspector:ListAssessmentRuns",

"inspector:ListAssessmentTargets",

"inspector:ListAssessmentTemplates",

"inspector:ListExclusions",

"inspector:ListFindings",

"inspector:PreviewAgents",

"kinesis:ListStreams",

"kinesis:DescribeStream",

"kinesis:ListTagsForStream",

"macie2:GetClassificationExportConfiguration",

"macie2:GetFindingStatistics",

"macie2:ListClassificationJobs",

"organizations:DescribeAccount",

"organizations:DescribeCreateAccountStatus",

"organizations:DescribeHandshake",

"organizations:DescribeOrganization",

"organizations:DescribeOrganizationalUnit",

"organizations:DescribePolicy",

"organizations:ListAWSServiceAccessForOrganization",

"organizations:ListAccounts",

"organizations:ListAccountsForParent",

"organizations:ListChildren",

"organizations:ListCreateAccountStatus",

"organizations:ListHandshakesForAccount",

"organizations:ListHandshakesForOrganization",

"organizations:ListOrganizationalUnitsForParent",

"organizations:ListParents",

"organizations:ListPolicies",

"organizations:ListPoliciesForTarget",

"organizations:ListRoots",

"organizations:ListTargetsForPolicy",

"rds:DescribeDBClusterParameters",

"rds:DescribeDBClusterParameterGroups",

"route53domains:GetDomainDetail",

"s3:GetAccelerateConfiguration",

"s3:GetAccountPublicAccessBlock",

"s3:GetBucketAcl",

"s3:GetBucketLocation",

"s3:GetBucketLogging",

"s3:GetBucketObjectLockConfiguration",

"s3:GetBucketPolicy",

"s3:GetBucketPolicyStatus",

"s3:GetBucketPublicAccessBlock",

"s3:GetBucketTagging",

"s3:GetBucketVersioning",

"s3:GetBucketWebsite",

"s3:GetEncryptionConfiguration",

"s3:GetLifecycleConfiguration",

"s3:ListBucket",

"s3:ListAllMyBuckets",

"securityhub:DescribeHub",

"securityhub:GetEnabledStandards",

"securityhub:GetFindings",

"securityhub:GetInsightResults",

"securityhub:GetInsights",

"securityhub:GetMasterAccount",

"securityhub:GetMembers",

"securityhub:ListEnabledProductsForImport",

"securityhub:ListInvitations",

"securityhub:ListMembers",

"servicequotas:ListServiceQuotas",

"sagemaker:DescribeNotebookInstance",

"sagemaker:ListNotebookInstances",

"sagemaker:ListTags",

"secretsmanager:DescribeSecret",

"secretsmanager:ListSecrets",

"shield:DescribeSubscription",

"ssm:DescribeParameters",

"ssm:DescribeSessions",

"ssm:DescribeInstanceInformation",

"storagegateway:DescribeNFSFileShares",

"storagegateway:DescribeSMBFileShares",

"storagegateway:DescribeTapes",

"storagegateway:ListFileShares",

"storagegateway:ListTagsForResource",

"storagegateway:ListTapes",

"transfer:DescribeServer",

"transfer:ListServers",

"xray:GetEncryptionConfig",

"waf:GetWebACL",

"waf:ListWebACLs",

"wafv2:ListWebACLs",

"workspaces:DescribeTags",

"workspaces:DescribeWorkspaces",

"workspaces:DescribeWorkspacesConnectionStatus",

"support:DescribeSeverityLevels",

"support:DescribeTrustedAdvisorChecks",

"support:DescribeTrustedAdvisorCheckResult",

"support:DescribeTrustedAdvisorCheckRefreshStatuses",

"support:RefreshTrustedAdvisorCheck",

"comprehend:ListKeyPhrasesDetectionJobs",

"comprehend:ListSentimentDetectionJobs",

"comprehend:ListTopicsDetectionJobs",

"comprehend:ListEntitiesDetectionJobs",

```
      "comprehend:ListDocumentClassificationJobs",
      "comprehend:ListDominantLanguageDetectionJobs",
      "wellarchitected:ListWorkloads",
      "wellarchitected:GetWorkload",
      "ecs:DescribeTaskDefinition",
      "ecs:ListTaskDefinitions",
      "compute-optimizer:GetAutoScalingGroupRecommendations",
      "compute-optimizer:GetEC2InstanceRecommendations",
      "ecs:ListClusters",
      "ecs:ListServices",
      "ecs:DescribeServices",
      "ecs:ListContainerInstances",
      "ecs:DescribeContainerInstances",
      "ecs:DescribeClusters",
      "ecs:ListTagsForResource",
      "appflow:DescribeFlow",
      "appflow:ListFlows"
    ],
    "Resource": [
      "*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3:Get*",
      "s3:List*"
    ],
    "Resource": "arn:aws:s3:::elasticbeanstalk*"
  }
]
```

```
    }
   }
  },
 "CloudConformityRole": {
   "Type": "AWS::IAM::Role",
   "Properties": {
     "RoleName": "CloudConformity",
     "AssumeRolePolicyDocument": {
       "Version": "2012-10-17",
       "Statement": [
        {
          "Effect": "Allow",
          "Principal": {
            "AWS": {
              "Fn::Join": [
                "",
                [
                  "arn:aws:iam::",
                  {
                    "Ref": "AccountId"
                  },
                  ":root"
                ]
              ]
            }
          },
          "Action": "sts:AssumeRole",
          "Condition": {
            "StringEquals": {
              "sts:ExternalId": {
                "Ref": "ExternalId"
```

```
            }
           }
          }
         }
        ]
       },
       "ManagedPolicyArns": [
        {
         "Ref": "CloudConformityCustomPolicyPart1"
        },
        {
         "Ref": "CloudConformityCustomPolicyPart2"
        }
       ]
      }
     }
    },
    "Outputs": {
     "CloudConformityRoleArn": {
      "Value": {
       "Fn::GetAtt": [
        "CloudConformityRole",
        "Arn"
       ]
      }
     },
     "Version": {
      "Value": "1.48"
     }
    }
}
```