Enhancing Cloud Security with CSPM Tools

Aligning Cybersecurity Strategies with SDG 16 for Stronger Institutions

SRM TRP ENGINEERING COLLEGE
Affiliated to ANNA UNIVERSITY
TIRUCHIRAPPALLI

Study of Cloud Security Posture Management (CSPM) Tools

Aligned with SDG 16: Peace, Justice, and Strong Institutions

Name: Nowshika Mirza. R

Roll Number: 814723104109

Course code: CCS335

Course name: Cloud computing

Department: CSE – B

Year: III

Project Overview

Cloud Security Posture Management (CSPM) tools are crucial for automating the detection, monitoring, and remediation of misconfigurations within cloud systems. These tools provide essential visibility into cloud assets, evaluate compliance, and ensure adherence to established security frameworks.

According to Data Insights Market projections for 2025, the CSPM market, currently valued at USD 10 billion, is expected to reach USD 40 billion by 2033. This growth is primarily driven by the adoption of multi-cloud environments and the increasing frequency of data breaches.

Common challenges faced by organizations, such as human error, inadequate encryption, and configuration drift, make CSPM tools indispensable. These challenges also highlight CSPM's alignment with SDG 16, which focuses on justice, integrity, and the establishment of strong digital institutions.

CSPM Tool Comparison Table

CSPM Tool	AI Remediation	Compliance Automation	Visibility Scope	Notable Feature
CloudDefense.AI	Yes	Yes	Multi-cloud	AI-based threat detection
Prisma Cloud (Palo Alto)	Yes	Yes	Full-stack	Risk analytics dashboard
Wiz	Yes	Yes	Multi-cloud	Agentless scanning
Orca Security	Yes	Yes	Broad	Context-aware risk prioritization
Check Point CloudGuard	No	Yes	Cloud-native	Policy enforcement engine
CrowdStrike Falcon	Yes	Partial	Hybrid	Runtime protection layer

Market Statistics and Graph Overview

CSPM Adoption (2025)

As per the Fortinet Cloud Security Report (2025):

- **67%** of organizations have implemented or plan to adopt CSPM.
- **52%** are integrating AI-based monitoring into posture management.
- Only **8%** encrypt more than 80% of their cloud data, emphasizing the need for compliance.

Graph Summary

A bar chart depicting CSPM adoption levels by organization size would show:

- **Startups**: 42%
- **SMBs**: 63%
- **Enterprises**: 78%
- **Government agencies**: 85%

Interpretation: The highest CSPM adoption rate is evident in governmental and financial institutions, highlighting their regulatory environment and alignment with SDG 16's emphasis on governance transparency.

CSPM Market Growth Data

Parameter	2025 Value	2034 Forecast	Growth Rate
Global CSPM Market	USD 6.43 Billion	USD 15.64 Billion	CAGR 10.37%
North American Market Share	40%	42%	Steady Rise
Asia-Pacific Region Growth	25% CAGR	Rapid Expansion	Increased regulatory frameworks

Insight

The strongest growth is seen in the Asia-Pacific region, driven by India's and China's adoption of data protection laws, which necessitate CSPM integration in financial, IT, and healthcare sectors.

Compliance Mapping Table

Framework	CSPM Relevance	Automated Audit Support
ISO 27001	Ensures configuration control for information security	Yes
GDPR	Detects unauthorized access to personal data	Yes
SOC 2	Tracks security controls for data integrity	Partial
NIST CSF	Aligns with cloud monitoring & alerting modules	Yes
HIPAA	Protects medical data integrity	Yes

Findings and Graph Insight

A comparative line graph (2020–2025 growth trend) reveals a significant increase in CSPM usage post-2022, correlating directly with the enactment of data privacy laws and major cloud breaches (e.g., Capital One).

Interpretation: The rise in CSPM adoption indicates an increased awareness of institutional accountability within cloud-reliant organizations.

Alignment with SDG 16

CSPM tools uphold SDG 16 by:

- **Promoting Transparency**: Continuous cloud monitoring ensures audit readiness.
- **Enforcing Accountability**: Automated compliance reporting enhances institutional governance.
- **Building Trustworthy Systems**: Prevents misuse, fraud, or corruption in digital platforms.

By automating controls and ensuring secure governance frameworks, CSPM tools support the principles of justice, peace, and strong institutions, fulfilling SDG 16 targets.

Key Features:

- Cleans and parses CloudTrail logs from CSV
- Flags misconfigurations and suspicious activities (ConsoleLogins, S3 ACL changes, etc.)
- Uses **Isolation Forest** to detect anomalous IP behavior
- Computes a composite risk score for each event and classifies levels (Low → Critical)
- Summarizes AWS region compliance percentages
- Generates **five interactive visualizations** using Plotly

Code:

Preview Code Blame 9129 lines (9129 loc) · 266 KB Raw

In [1]:
```
pip install pandas numpy matplotlib seaborn plotly scikit-learn folium geoip2 tqdm openpyxl
```

```
Requirement already satisfied: pandas in /opt/homebrew/anaconda3/lib/python3.11/site-packages (2.1.4)
Requirement already satisfied: numpy in /opt/homebrew/anaconda3/lib/python3.11/site-packages (1.26.2)
Requirement already satisfied: matplotlib in /opt/homebrew/anaconda3/lib/python3.11/site-packages (3.8.0)
Requirement already satisfied: seaborn in /opt/homebrew/anaconda3/lib/python3.11/site-packages (0.12.2)
Requirement already satisfied: plotly in /opt/homebrew/anaconda3/lib/python3.11/site-packages (5.9.0)
Requirement already satisfied: scikit-learn in /opt/homebrew/anaconda3/lib/python3.11/site-packages (1.2.2)
Requirement already satisfied: folium in /opt/homebrew/anaconda3/lib/python3.11/site-packages (0.20.0)
Collecting geoip2
  Downloading geoip2-5.1.0-py3-none-any.whl.metadata (19 kB)
Requirement already satisfied: tqdm in /opt/homebrew/anaconda3/lib/python3.11/site-packages (4.65.0)
Requirement already satisfied: openpyxl in /opt/homebrew/anaconda3/lib/python3.11/site-packages (3.0.10)
Requirement already satisfied: python-dateutil>=2.8.2 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (fro
m pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from pandas)
(2023.3.post1)
Requirement already satisfied: tzdata>=2022.1 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from panda
s) (2023.3)
Requirement already satisfied: contourpy>=1.0.1 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from matp
lotlib) (1.2.0)
Requirement already satisfied: cycler>=0.10 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from matplotl
ib) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from mat
plotlib) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from mat
plotlib) (1.4.4)
Requirement already satisfied: packaging>=20.0 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from matpl
otlib) (23.1)
Requirement already satisfied: pillow>=6.2.0 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from matplot
lib) (10.0.1)
Requirement already satisfied: pyparsing>=2.3.1 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from matp
lotlib) (3.0.9)
Requirement already satisfied: tenacity>=6.2.0 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from plotl
y) (8.2.2)
Requirement already satisfied: scipy>=1.3.2 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from scikit-l
earn) (1.16.0)
Requirement already satisfied: joblib>=1.1.1 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from scikit-
learn) (1.2.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from
scikit-learn) (2.2.0)
Requirement already satisfied: branca>=0.6.0 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from folium)
(0.8.2)
Requirement already satisfied: jinja2>=2.9 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from folium)
```

CSPM_CloudTrail_Simulation / CSPM Mini-Project Cloud Computing.ipynb ↑ Top

Preview Code Blame 9129 lines (9129 loc) · 266 KB Raw

```
(2022.9.0)
Requirement already satisfied: aiohttp<4.0.0,>=3.6.2 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from
geoip2) (3.9.0)
Collecting maxminddb<3.0.0,>=2.7.0 (from geoip2)
  Downloading maxminddb-2.8.2-cp311-cp311-macosx_11_0_arm64.whl.metadata (5.1 kB)
Requirement already satisfied: et_xmlfile in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from openpyxl)
(1.1.0)
Requirement already satisfied: attrs>=17.3.0 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from aiohttp
<4.0.0,>=3.6.2->geoip2) (23.1.0)
Requirement already satisfied: multidict<7.0,>=4.5 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from a
iohttp<4.0.0,>=3.6.2->geoip2) (6.0.4)
Requirement already satisfied: yarl<2.0,>=1.0 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from aiohtt
p<4.0.0,>=3.6.2->geoip2) (1.9.3)
Requirement already satisfied: frozenlist>=1.1.1 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from aio
http<4.0.0,>=3.6.2->geoip2) (1.4.0)
Requirement already satisfied: aiosignal>=1.1.2 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from aioh
ttp<4.0.0,>=3.6.2->geoip2) (1.2.0)
Requirement already satisfied: MarkupSafe>=2.0 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from jinja
2>=2.9->folium) (2.1.1)
Requirement already satisfied: six>=1.5 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from python-dateu
til>=2.8.2->pandas) (1.16.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (f
rom requests->folium) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from requests
->folium) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from re
quests->folium) (1.26.18)
Requirement already satisfied: certifi>=2017.4.17 in /opt/homebrew/anaconda3/lib/python3.11/site-packages (from re
quests->folium) (2024.2.2)
Downloading geoip2-5.1.0-py3-none-any.whl (27 kB)
Downloading maxminddb-2.8.2-cp311-cp311-macosx_11_0_arm64.whl (35 kB)
Installing collected packages: maxminddb, geoip2
Successfully installed geoip2-5.1.0 maxminddb-2.8.2

[notice] A new release of pip is available: 24.0 -> 25.2
[notice] To update, run: /opt/homebrew/anaconda3/bin/python -m pip install --upgrade pip
Note: you may need to restart the kernel to use updated packages.
```

In [3]:

```
# ================================================================
# AWS CloudTrail Mini CSPM Project (GitHub-Friendly Version)
# Author: Nowshika Mirza R
# Dataset: AWS CloudTrails Dataset from flaws.cloud (Kaggle)
# Purpose: Use minimal CSV sample for demonstration & GitHub upload
# ================================================================
```

CSPM_CloudTrail_Simulation / CSPM Mini-Project Cloud Computing.ipynb

↑ Top

Preview | Code | Blame | 9129 lines (9129 loc) · 266 KB

Raw

In [3]:

```
# ==================================================================
# AWS CloudTrail Mini CSPM Project (GitHub-Friendly Version)
# Author: Nowshika Mirza R
# Dataset: AWS CloudTrails Dataset from flaws.cloud (Kaggle)
# Purpose: Use minimal CSV sample for demonstration & GitHub upload
# ==================================================================

# Install dependencies (only if not already installed)
# !pip install pandas numpy matplotlib seaborn plotly scikit-learn tqdm

# ==================================================================
# 1. Imports & Initial Config
# ==================================================================
import os, json
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
from sklearn.ensemble import IsolationForest
from sklearn.preprocessing import StandardScaler
from tqdm import tqdm

sns.set(style="whitegrid")

CONFIG = {
    "INPUT_CSV": "cloudtrail_logs.csv",
    "OUTPUT_DIR": "cspm_outputs",
    "ANOMALY_RANDOM_STATE": 42,
    "ISOLATION_FOREST_CONTAMINATION": 0.05
}
os.makedirs(CONFIG["OUTPUT_DIR"], exist_ok=True)

# ==================================================================
# 2. Create Mini Dataset from Kaggle File
# ==================================================================
# If you download the Kaggle file (dec12_18features.csv) and keep it here,
# this cell creates a small subset safe for GitHub.
FULL_FILE = "dec12_18features.csv"
SAMPLE_FILE = CONFIG["INPUT_CSV"]

if os.path.exists(FULL_FILE):
    df_small = pd.read_csv(FULL_FILE, nrows=500)  # only first 500 rows
    df_small.to_csv(SAMPLE_FILE, index=False)
```

CSPM_CloudTrail_Simulation / CSPM Mini-Project Cloud Computing.ipynb

Preview Code Blame 9129 lines (9129 loc) · 266 KB

```python
if os.path.exists(FULL_FILE):
    df_small = pd.read_csv(FULL_FILE, nrows=500)  # only first 500 rows
    df_small.to_csv(SAMPLE_FILE, index=False)
    print(f"Sample created and saved as: {SAMPLE_FILE}")
else:
    print("⚠️ Kaggle full dataset not found. Please place 'dec12_18features.csv' in this folder.")

# ================================================================
# 3. Load Sample Dataset
# ================================================================
if os.path.exists(SAMPLE_FILE):
    df = pd.read_csv(SAMPLE_FILE)
else:
    df = pd.DataFrame({
        "eventTime": pd.date_range("2025-01-01", periods=50, freq="H"),
        "eventName": np.random.choice(["ConsoleLogin","CreateUser","PutBucketAcl"], 50),
        "awsRegion": np.random.choice(["us-east-1","eu-west-1"], 50),
        "sourceIPAddress": np.random.choice(["192.168.1.1","54.12.123.10"], 50),
        "errorCode": np.random.choice([None,"AccessDenied"], 50)
    })
    df.to_csv(SAMPLE_FILE, index=False)
    print(f"No Kaggle data found — demo dataset '{SAMPLE_FILE}' generated.")

print("Dataset loaded:", len(df), "rows")

# ================================================================
# 4. Rule-Based Flags (CSPM simulation)
# ================================================================
df["suspicious_event"] = df["eventName"].isin(["ConsoleLogin","CreateUser","PutBucketAcl"])
df["failed_login"] = df["eventName"].eq("ConsoleLogin") & df["errorCode"].notna()
df["rule_score"] = df[["suspicious_event","failed_login"]].sum(axis=1)

# ================================================================
# 5. Risk Scoring & Simple Anomaly Detection
# ================================================================
df["risk_score"] = df["rule_score"]*10 + df["errorCode"].notna()*20
agg = df.groupby("sourceIPAddress").agg(event_count=("eventName","count")).reset_index()
scaler = StandardScaler()
iso = IsolationForest(contamination=CONFIG["ISOLATION_FOREST_CONTAMINATION"],
                      random_state=CONFIG["ANOMALY_RANDOM_STATE"])
agg["scaled"] = iso.fit_predict(scaler.fit_transform(agg[["event_count"]]))
agg["ip_anomaly"] = agg["scaled"]==-1
df = df.merge(agg[["sourceIPAddress","ip_anomaly"]], on="sourceIPAddress", how="left")
df["risk_score"] += df["ip_anomaly"].astype(int)*30
```

CSPM_CloudTrail_Simulation / CSPM Mini-Project Cloud Computing.ipynb

Preview Code Blame 9129 lines (9129 loc) · 266 KB

```python
agg["scaled"] = iso.fit_predict(scaler.fit_transform(agg[["event_count"]]))
agg["ip_anomaly"] = agg["scaled"]==-1
df = df.merge(agg[["sourceIPAddress","ip_anomaly"]], on="sourceIPAddress", how="left")
df["risk_score"] += df["ip_anomaly"].astype(int)*30

def classify(score):
    if score>=60: return "Critical"
    elif score>=40: return "High"
    elif score>=20: return "Medium"
    return "Low"
df["Risk_Level"] = df["risk_score"].apply(classify)

# ================================================================
# 6. Basic Reporting
# ================================================================
summary = df["Risk_Level"].value_counts().reset_index()
summary.columns = ["Risk_Level","Count"]
print(summary)

summary.to_csv(os.path.join(CONFIG["OUTPUT_DIR"],"risk_summary.csv"), index=False)
df.to_csv(os.path.join(CONFIG["OUTPUT_DIR"],"annotated_cloudtrail.csv"), index=False)

fig = px.pie(summary, names="Risk_Level", values="Count",
             title="Risk Level Distribution (Mini Sample)",
             color="Risk_Level",
             color_discrete_map={"Low":"green","Medium":"orange","High":"red","Critical":"darkred"})
fig.show()

print("\nAll outputs saved in:", CONFIG["OUTPUT_DIR"])
```

CSPM_CloudTrail_Simulation / CSPM Mini-Project Cloud Computing.ipynb

↑ Top

Preview | Code | Blame | 9129 lines (9129 loc) · 266 KB | Raw

In [4]:

```
# ================================================================
# AWS CloudTrail CSPM Mini Project — Full GitHub-Size Edition
# Author: Nowshika Mirza R
# ================================================================

# !pip install pandas numpy matplotlib seaborn plotly scikit-learn tqdm

# ================================================================
# 1. Setup & Imports
# ================================================================
import os, json, numpy as np, pandas as pd
import matplotlib.pyplot as plt, seaborn as sns, plotly.express as px
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import IsolationForest
from tqdm import tqdm

sns.set(style="whitegrid")

CONFIG = {
    "INPUT_CSV": "cloudtrail_logs_sample.csv",
    "OUTPUT_DIR": "cspm_outputs",
    "ISOLATION_FOREST_CONTAMINATION": 0.05,
    "ANOMALY_RANDOM_STATE": 42
}
os.makedirs(CONFIG["OUTPUT_DIR"], exist_ok=True)

# ================================================================
# 2. Create / Load Lightweight Dataset
# ================================================================
FULL_FILE = "dec12_18features.csv"
SAMPLE_FILE = CONFIG["INPUT_CSV"]

if os.path.exists(FULL_FILE):
    df_small = pd.read_csv(FULL_FILE, nrows=800)
    df_small.to_csv(SAMPLE_FILE, index=False)
    print(f"Sample saved as {SAMPLE_FILE}")
else:
    print("⚠️ Please place Kaggle file dec12_18features.csv here first.")

df = pd.read_csv(SAMPLE_FILE)
print("Rows:", len(df), "Columns:", df.columns.tolist()[:10])

# ================================================================
# 3. Basic Cleaning
```

CSPM_CloudTrail_Simulation / CSPM Mini-Project Cloud Computing.ipynb

Preview Code Blame 9129 lines (9129 loc) · 266 KB Raw

```
# ==================================================================
# 3. Basic Cleaning
# ==================================================================
df.rename(columns=lambda x: x.strip(), inplace=True)
if "eventTime" in df.columns:
    df["eventTime"] = pd.to_datetime(df["eventTime"], errors="coerce")

core = ["eventTime","eventName","awsRegion","sourceIPAddress","errorCode"]
for c in core:
    if c not in df.columns:
        df[c] = None
df.dropna(subset=["eventName"], inplace=True)

# ==================================================================
# 4. CSPM-Style Rule Flags
# ==================================================================
df["suspicious_event"] = df["eventName"].isin(
    ["ConsoleLogin","AuthorizeSecurityGroupIngress","AttachRolePolicy",
     "CreateUser","DeleteBucket","PutBucketAcl","PutBucketPolicy"]
)
df["failed_login"] = (df["eventName"]=="ConsoleLogin") & df["errorCode"].notna()
df["rule_score"] = df[["suspicious_event","failed_login"]].sum(axis=1)

# ==================================================================
# 5. Risk Scoring & Anomaly Detection (IsolationForest)
# ==================================================================
df["risk_score"] = df["rule_score"]*10 + df["errorCode"].notna()*20
agg = df.groupby("sourceIPAddress").agg(event_count=("eventName","count")).reset_index()
scaler = StandardScaler()
iso = IsolationForest(random_state=CONFIG["ANOMALY_RANDOM_STATE"],
                      contamination=CONFIG["ISOLATION_FOREST_CONTAMINATION"])
agg["flag"] = iso.fit_predict(scaler.fit_transform(agg[["event_count"]]))
agg["ip_anomaly"] = agg["flag"]==-1
df = df.merge(agg[["sourceIPAddress","ip_anomaly"]], on="sourceIPAddress", how="left")
df["risk_score"] += df["ip_anomaly"].astype(int)*30

def classify(score):
    if score>=60: return "Critical"
    elif score>=40: return "High"
    elif score>=20: return "Medium"
    return "Low"
df["Risk_Level"] = df["risk_score"].apply(classify)

# ==================================================================
# 6. Region-wise Compliance Simulation
```

CSPM_CloudTrail_Simulation / CSPM Mini-Project Cloud Computing.ipynb

Preview | Code | Blame | 9129 lines (9129 loc) · 266 KB | Raw

```python
# ==============================================================
# 6. Region-wise Compliance Simulation
# ==============================================================
df["Compliant"] = ~((df["eventName"].isin(["PutBucketAcl","PutBucketPolicy"])) |
                    (df["errorCode"].notna()))

region_summary = df.groupby("awsRegion").agg(
    events=("eventName","count"),
    avg_risk=("risk_score","mean"),
    compliance_pct=("Compliant", lambda x: x.mean()*100)
).reset_index()

# ==============================================================
# 7. Save Key Outputs
# ==============================================================
os.makedirs(CONFIG["OUTPUT_DIR"], exist_ok=True)
df.to_csv(os.path.join(CONFIG["OUTPUT_DIR"],"annotated_cloudtrail.csv"), index=False)
region_summary.to_csv(os.path.join(CONFIG["OUTPUT_DIR"],"region_summary.csv"), index=False)

print("All processed files saved to:", CONFIG["OUTPUT_DIR"])

# ==============================================================
# 8. Visualizations (5 charts)
# ==============================================================
# 8.1 Risk Level Pie
fig1 = px.pie(df, names="Risk_Level", title="Overall Risk Distribution",
              color="Risk_Level",
              color_discrete_map={"Low":"green","Medium":"orange","High":"red","Critical":"darkred"})
fig1.show()

# 8.2 Top Event Types by Frequency & Risk
top_events = df.groupby("eventName").agg(count=("eventName","count"),
                                         avg_risk=("risk_score","mean")).reset_index().sort_values("count",ascenc
fig2 = px.bar(top_events, x="eventName", y="count", color="avg_risk",
              title="Top Event Types (by Count and Average Risk)", color_continuous_scale="Reds")
fig2.update_layout(xaxis_tickangle=-45)
fig2.show()

# 8.3 Timeline of Events per Day
if "eventTime" in df.columns:
    tl = df.set_index("eventTime").resample("D").size().reset_index(name="daily_events")
    fig3 = px.line(tl, x="eventTime", y="daily_events",
                   title="Events Per Day (Timeline)")
    fig3.show()
```

cloudtrail_logs_sample.csv:

Preview | Code | Blame | 801 lines (801 loc) · 199 KB | Raw

Search this file

	eventID	eventTime	sourceIPAddress	userAgent	eventName	eventSource
2	3038ebd2-c98a-4c65-9b6e-e22506292313	2017-02	255.253	[S3Console	ListBuckets	s3.amazonaws.com
3	22a0d9b1-deea-4d39-827b-2af7050ed3f3	2017-02	255.253	console.amazonaws.com	GetAccountPasswordPolicy	iam.amazonaws.com
4	9facf7ca-cb76-4b19-940c-3de6803f7efb	2017-02	255.253	console.amazonaws.com	GetAccountSummary	iam.amazonaws.com
5	6596d3b4-7c98-40b1-867d-f317f1dbdc18	2017-02	255.253	console.amazonaws.com	ListAccountAliases	iam.amazonaws.com
6	9f9d038c-e5a5-443e-83d5-4cf00941d399	2017-02	255.253	console.amazonaws.com	ListMFADevices	iam.amazonaws.com
7	4babc3a3-77b1-44b6-9940-42363d44f5b2	2017-02	255.253	console.amazonaws.com	ListAccessKeys	iam.amazonaws.com
8	c2f959326-973c-4508-bc82-5ec06bea252f	2017-02	255.253	console.amazonaws.com	ListAccessKeys	iam.amazonaws.com
9	eec27e8a-b750-4c7c-95e3-99a80d203bee	2017-02	255.253	console.amazonaws.com	GetAccountPasswordPolicy	iam.amazonaws.com
10	30f077e5-6c11-41f6-8ed0-2aedad506b1d	2017-02	255.253	console.amazonaws.com	GetAccountSummary	iam.amazonaws.com
11	dc97bcac-91e6-490f-9f7c-32e56cc58dcd	2017-02	255.253	console.amazonaws.com	ListAccountAliases	iam.amazonaws.com
12	e5cacd34-cfa7-4b74-a380-51dbb56832a	2017-02	255.253	console.amazonaws.com	ListMFADevices	iam.amazonaws.com
13	09bca6e3-ff6c-49f5-8735-d4389b10c769	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
14	2dd9d57f-2734-49e4-a801-ab9a7fb21dd3	2017-02	Unknown	[AWSCloudTrail, aws-internal	ListBuckets	s3.amazonaws.com
15	032db777-fcc7-4fdd-a916-c60e1050db97	2017-02	255.253	AWSCloudTrail, aws-internal	ListTopics	sns.amazonaws.com
16	5e41180-aa5c-4599-89f1-1eaf8c77389c	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
17	ff9cea3c-940e-4eb7-863e-df7d4ff618a8	2017-02	Unknown	[AWSCloudTrail, aws-internal	CreateBucket	s3.amazonaws.com
19	d7d9f9c9-a19b-417c-a925-a378150b8640	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
20	407b5157-e6e2-46de-854a-dca35b5a2697	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
21	d6633d9e-73cb-433b-8016-7b65bc970805d	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
22	976a7a83-ada1-48f7-bf14-c60eaf4f1ccb	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
23	a3f14f50-0b0c-407b-9cbd-62f4300b2c5e	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
24	e6fa7d52-cea1-44c7-8bef-7d065aa19aff	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
25	85475f09-add7-48e5-80a5-3199f6a45937	2017-02	255.253	console.amazonaws.com	CreateTrail	cloudtrail.amazonaws.com
26	d184113-6f9c-4340-9c9d-b456636ba3d5	2017-02	255.253	console.amazonaws.com	StartLogging	cloudtrail.amazonaws.com
27	08b62c21-0bf2-4b65-9a52-1033f7f937313	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
28	2fceb21a-9120-4ab5-9d39-d62a6f235820	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
29	b859216ba-2710-4127-b04c-4ed0eac3ef99	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
30	708788a3-55a6-41d6-90d2-0170a282f29a	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
31	17f0abda-ea87-4005-94d1-6c302a52897	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
32	e98ed1de-066a-4bdb-a67f-0d34a0c7ace8	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
33	f66379a9-4fa4-4b69-a028-d7111117f8808e	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
34	a9176e03-33c4-4048-94b0-8a88e8eded3c	2017-02	255.253	console.amazonaws.com	DescribeTrails	cloudtrail.amazonaws.com
35	58b59f01-36e8-43cc-be97-35e8b3a72ea5	2017-02	255.253	console.amazonaws.com	GetTrailStatus	cloudtrail.amazonaws.com
36	37e9bbe2-82a2-413c-95a2-e339b5ead4a8	2017-02	255.253	[S3Console	ListBuckets	s3.amazonaws.com
37	0ceeba88-3e7f-4b35-800d-1d59bca73e2e	2017-02	85.50	console.ec2.amazonaws.com	DescribeInstances	ec2.amazonaws.com

38	4a405c8c-150f-435a-97fa-b36ec1a2da06	2017-02	85.50	console.ec2.amazonaws.com	DescribeAvailabilityZones	ec2.amazonaws.com
39	5d5f2ede-af65-42a3-bfde-0cbd74b0ba9b	2017-02	85.50	console.ec2.amazonaws.com	DescribeLoadBalancers	elasticloadbalancing.amazona
40	d4ff874b-dfb7-4d1b-9e27-ca6948e5606b	2017-02	85.50	console.ec2.amazonaws.com	DescribeAvailabilityZones	ec2.amazonaws.com
41	332d8a85-8e0e-4aa5-a6be-f49be9ada643	2017-02	85.50	console.ec2.amazonaws.com	DescribeTags	ec2.amazonaws.com
42	f0df16e8-b423-4cbe-b46f-0c95f65daebe	2017-02	85.50	console.ec2.amazonaws.com	DescribeSnapshots	ec2.amazonaws.com
43	37d276de-e12a-446b-a5cc-d8a2f849f44e	2017-02	85.50	console.ec2.amazonaws.com	DescribeVolumeStatus	ec2.amazonaws.com
44	4c4e8124-818d-4c7f-b6bf-ae65515a7cd9	2017-02	85.50	console.ec2.amazonaws.com	DescribeVolumes	ec2.amazonaws.com
45	571b8acc-01fe-4472-8538-e50a51f41dff	2017-02	85.50	console.ec2.amazonaws.com	DescribeInstanceStatus	ec2.amazonaws.com
46	6c3b8eb4-cbf3-401a-bd4d-d74a08a0480d	2017-02	85.50	console.ec2.amazonaws.com	DescribeVolumeStatus	ec2.amazonaws.com
47	cb9a1f88-bc6e-44c7-b17d-9619b5ab3c11	2017-02	85.50	console.ec2.amazonaws.com	DescribeVolumes	ec2.amazonaws.com
48	a562e806-55a6-49dd-a15d-ff6ec41a1159	2017-02	85.50	console.ec2.amazonaws.com	DescribeVolumes	ec2.amazonaws.com
49	fd4fcecc-4fd9-4b2d-be53-e19b5386dc43	2017-02	85.50	console.ec2.amazonaws.com	DescribeVolumeStatus	ec2.amazonaws.com
50	4f14cb58-f5a0-44e9-b3ec-e4dff97d81db	2017-02	255.253	console.ec2.amazonaws.com	DescribeVolumeStatus	ec2.amazonaws.com
51	c41ce463-4aad-4476-bec3-5c76609efc	2017-02	255.253	console.ec2.amazonaws.com	DescribeAddresses	ec2.amazonaws.com
52	595dbd59-344a-4c4f-bda6-6820cc0ffd4b	2017-02	255.253	console.ec2.amazonaws.com	DescribeSnapshots	ec2.amazonaws.com
53	c524273-3236-438d-92a0-3a0503c1f524	2017-02	255.253	console.ec2.amazonaws.com	DescribeAvailabilityZones	ec2.amazonaws.com
54	3b97792-f363-4ac7-bdb9-eb3045d96e5c	2017-02	255.253	console.ec2.amazonaws.com	DescribeAccountAttributes	ec2.amazonaws.com
55	49eecac4-30ee-4337-96fc-d7c284e557cd	2017-02	255.253	console.ec2.amazonaws.com	DescribeKeyPairs	ec2.amazonaws.com
56	79d382e6-400d-4799-88f8-1642fd882d06	2017-02	255.253	console.ec2.amazonaws.com	DescribeHosts	ec2.amazonaws.com
57	0ee520038-f22c-4daa-8582-4c105ce8d676	2017-02	255.253	console.ec2.amazonaws.com	DescribeInstances	ec2.amazonaws.com

61	79d5edf2-eb2f-4dc9-a21c-1878c64580f1	2017-02	255.253	console.ec2.amazonaws.com	DescribePlacementGroups	ec2.amazonaws.com
62	b627088ef-216a-4e89-9460-bd8d2aad75dc	2017-02	255.253	console.ec2.amazonaws.com	DescribeLoadBalancers	elasticloadbalancing.amazona
63	2f741b6b-af9a-4b7e-8628-97d698622	2017-02	255.253	console.ec2.amazonaws.com	DescribeAccountAttributes	ec2.amazonaws.com
64	728a819f-f84d-4e20-916b-f66f81990b16	2017-02	255.253	console.ec2.amazonaws.com	DescribeSecurityGroups	ec2.amazonaws.com
65	d74b234a-bdd1-457a-8758-afc57affd6ee	2017-02	255.253	console.ec2.amazonaws.com	DescribeSubnets	ec2.amazonaws.com
66	bab3dbc0-8df6-4a70-abbd-c7849d6a7295	2017-02	255.253	console.ec2.amazonaws.com	DescribeVpcClassicLink	ec2.amazonaws.com
67	8f2233af-a448-488a-b9f6-243998e2d8	2017-02	255.253	console.ec2.amazonaws.com	DescribeImages	ec2.amazonaws.com
68	9c73e463-7fec-435e-8c5b-d676b6603c44	2017-02	255.253	console.ec2.amazonaws.com	DescribeVpcs	ec2.amazonaws.com
69	a459c448-183f-4f35-a566-d00168fd10	2017-02	255.253	aws-internal	ListInstanceProfiles	iam.amazonaws.com
70	60cdf866-1318-4dd6-b55d-09950a32a62b	2017-02	255.253	console.ec2.amazonaws.com	DescribeHosts	ec2.amazonaws.com
71	326e33e9-8475-4cbf-9b52-2f5ff7638e80	2017-02	255.253	console.ec2.amazonaws.com	DescribeImages	ec2.amazonaws.com
72	004970-faac-4065-b040-35de20850c43	2017-02	255.253	Coral	DescribeDirectories	ds.amazonaws.com
73	8ce8799d-79ae-46c4-bb6f-18161ee2e9da	2017-02	255.253	console.ec2.amazonaws.com	DescribeSpotPriceHistory	ec2.amazonaws.com
74	70979d-b8a3-463a-8860-e8f85272e608	2017-02	255.253	console.ec2.amazonaws.com	DescribeNetworkInterfaces	ec2.amazonaws.com
75	bcb833703-c410-4d82-b759-758e2fe916a1	2017-02	255.253	console.ec2.amazonaws.com	DescribeSecurityGroups	ec2.amazonaws.com
76	82486ff-130f-43a6-b7c4-f8efa71af896	2017-02	255.253	console.ec2.amazonaws.com	DescribeAvailabilityZones	ec2.amazonaws.com
77	7ad47b8d-09be-4b40-8b0a-48e85f0bf123	2017-02	255.253	console.ec2.amazonaws.com	DescribeImages	ec2.amazonaws.com
78	c6557ca3-cc04-4e5f-b298-8d367b10e116	2017-02	255.253	console.ec2.amazonaws.com	DescribeSecurityGroups	ec2.amazonaws.com
79	48e533df-8f81-4d1c-a62a-c6ccb4ff3f9a	2017-02	255.253	console.ec2.amazonaws.com	DescribeLoadBalancers	elasticloadbalancing.amazona

Conclusion

CSPM tools are a cornerstone of modern cloud security. Through visualization dashboards, automated enforcement, and compliance integration, organizations can protect critical infrastructure, gain insight into security postures, and swiftly address misconfigurations.

As digital governance becomes increasingly pivotal worldwide, CSPM's role in ensuring transparent, resilient, and just institutions continues to expand, substantiating its alignment with the United Nations SDG 16.