

A Defensive Cloud and Drone Security Posture Assessment Framework for Multi-Cloud UAV Operations

CD-SPAF: Cloud-Drone Security Posture Assessment Framework

Shahriar Badhan

MSc Cyber Security and Digital Forensics

University of Wales Trinity Saint David (UWTSD), London Campus

Supervisor: Dr Dipo Dunsin

September 2026



Research Background



Simplified cloud-connected UAV data flow assumed in the dissertation (Fig. 1.1)



Why cloud security matters for UAVs

- UAVs are used surveillance, inspection, agriculture, emergency response and logistics [1]
- Cloud provides storage, analytics, identity, logging and fleet management
- Therefore, cloud security is important for UAV operations.



Security challenges

- UAV threats: communication interception, GNSS spoofing, jamming, denial of service [2]
- Cloud risks: weak IAM, insecure storage, excessive permissions, poor logging, weak device identity
- AWS and Azure differ in services and controls, so joint assessment is harder

Research problem: UAV security and cloud security are usually assessed separately. My research aims to combine them into one framework to identify and assess security risks.

Research Aim, Objectives and Question



Aim

To design, develop and evaluate a defensive security posture assessment framework for UAV operations that depend on cloud infrastructure across multiple cloud environments.



Research question

How can a single framework assess security risks in cloud-connected UAV operations using AWS and Azure?

Objectives

1

Identify security weaknesses in cloud-supported UAV architectures: IAM, storage, logging, device identity, network exposure and telemetry

2

Design the CD-SPAF framework using NIST CSF 2.0 and Zero Trust principles

3

Develop a python prototype that applies predefined security rules to selected cloud information and controlled UAV telemetry

4

Implement a risk assessment mechanism that categorises findings by severity

5

Evaluate the prototype in controlled scenarios, including the usefulness of its recommendations

Literature Review and Research Gap

Key findings from the review



UAV security research

Focuses on the aircraft, communications and navigation, such as GNSS spoofing [1], [2]



AWS and Azure guidance

Strong identity, detection and data-protection controls, but provider-specific [5], [6]



Cloud security posture management

Finds cloud configuration weaknesses, but has no UAV context [7]



NIST CSF 2.0 and Zero Trust

Give principles and structure, not a combined UAV-cloud workflow [3], [4]

Approaches compared (dissertation Table 2.2)

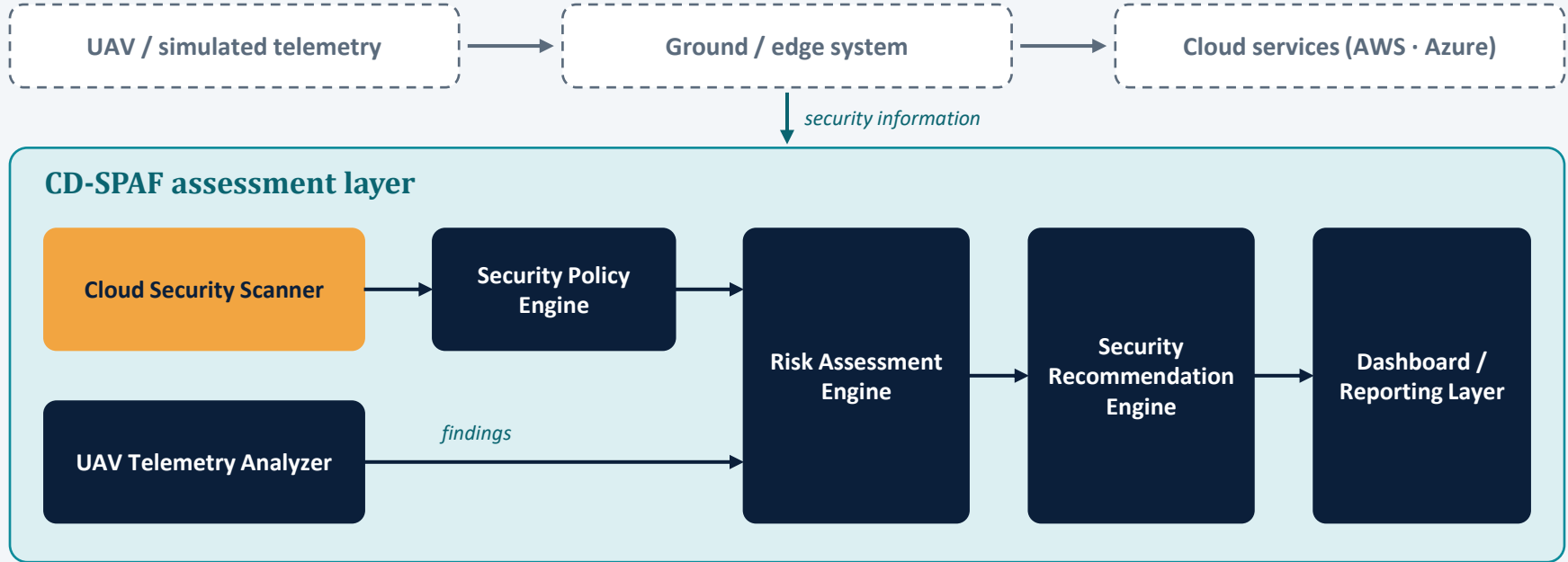
Approach	UAV awareness	Cloud posture	Unified risk view
UAV communication research	High	Low	Low
GNSS spoofing research	High	Low	Low
AWS security guidance	Low	High	Provider-specific
Azure / MCSB guidance	Low	High	Provider-oriented
CSPM approaches	Low	High	Moderate
Zero Trust Architecture	Medium	High	Low
CD-SPAF (proposed)	High	High	Yes

Ratings as reported in the dissertation's literature comparison, not experimental results.

Research gap: Previous research mainly focused on drone security and cloud security separately. My research aims to combine both in one framework to identify and assess security risks.

Proposed Framework: CD-SPAF

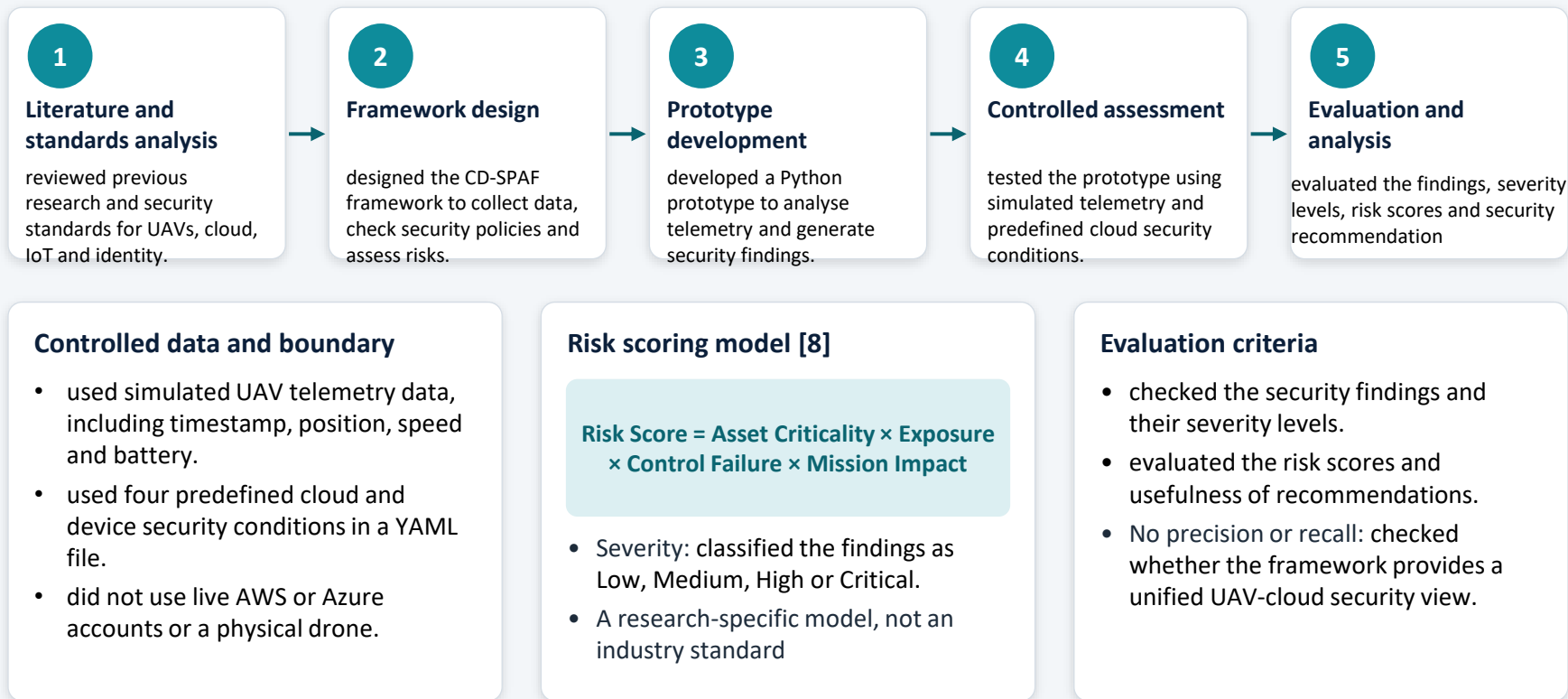
Operational UAV-cloud environment (simulated telemetry and predefined conditions in the prototype)



■ Implemented in prototype ■ Simulated input: predefined posture conditions, no live AWS / Azure API

Every finding shares one structure: finding · category · evidence · severity · risk score · recommendation

Research Methodology



Implementation

Component	Technology used
Main application	Python
Telemetry processing	Pandas, NumPy
Anomaly detection	scikit-learn: Isolation Forest, DBSCAN
Policy evaluation	Python with YAML rules
Risk and recommendations	Python (category mapping table)
Reporting	JSON report
Dashboard	Streamlit, Plotly Express

Implemented

- implemented telemetry anomaly detection using Python.
- developed a YAML-based security policy engine in demo mode.
- developed a Streamlit dashboard and performed unit testing

Not implemented

- Live AWS / Azure collectors
- Live or real-time UAV telemetry
- Physical drone testing
- Detection accuracy metrics

```
telemetry > detector.py
1 from __future__ import annotations
2 from pathlib import Path
3 from typing import Any
4 import numpy as np
5 import pandas as pd
6 from sklearn.ensemble import IsolationForest
7 from sklearn.cluster import DBSCAN
8
9 REQUIRED = ["timestamp", "drone_id", "latitude", "longitude", "altitude", "speed", "battery"]
10 NUMERIC = ["latitude", "longitude", "altitude", "speed", "battery"]
11
12 class TelemetryAnomalyDetector:
13     def load_csv(self, path: str | Path) -> pd.DataFrame:
14         df = pd.read_csv(path, parse_dates=["timestamp"])
15         missing = sorted(set(REQUIRED) - set(df.columns))
16         if missing:
17             raise ValueError(f"Missing columns: {missing}")
```

telemetry/detector.py

```
policy > engine.py
1 from __future__ import annotations
2 from pathlib import Path
3 from typing import Any
4 import yaml
5
6 class PolicyEngine:
7     def __init__(self, policy_file: Path):
8         self.policy_file = policy_file
9         self.policies = yaml.safe_load(policy_file.read_text())
10
11     def evaluate_demo(self) -> list[dict[str, Any]]:
12         findings = []
13         for policy in self.policies.get("policies", []):
14             if policy.get("demo_violation", False):
15                 findings.append({"finding_id": policy["id"], "text": policy["text"]})
16         return findings
```

policy/engine.py

```
risk > recommendations.py
1 from __future__ import annotations
2 from typing import Any
3
4 MAP = {
5     "Cloud Storage": ("NIST CSF 2.0": "PR.DS", "CIS AWS Benchmark": "1.1.1", "MITRE ATT&CK": "TA0001"),
6     "Cloud Identity": ("NIST CSF 2.0": "PR.AA", "CIS AWS Benchmark": "1.1.2", "MITRE ATT&CK": "TA0001"),
7     "IoT Identity": ("NIST CSF 2.0": "PR.AA", "OWASP IoT": "1.1.1", "MITRE ATT&CK": "TA0001"),
8     "Monitoring": ("NIST CSF 2.0": "DE.CH", "MITRE ATT&CK": "TA0001", "OWASP IoT": "1.1.1", "MITRE ATT&CK": "TA0001"),
9     "UAV Telemetry": ("NIST CSF 2.0": "DE.CH", "OWASP IoT": "1.1.1", "MITRE ATT&CK": "TA0001")
10 }
11
12 class RecommendationEngine:
13     def generate(self, findings: list[dict[str, Any]]) -> list[dict[str, Any]]:
14         out = {}
15         for f in findings:
16             mapping = MAP.get(f["category"], ("NIST CSF 2.0": "DE.CH", "OWASP IoT": "1.1.1", "MITRE ATT&CK": "TA0001"))
```

risk/recommendations.py

```
dashboard > app.py
1 from __future__ import annotations
2 from pathlib import Path
3 import json
4 import pandas as pd
5 import plotly.express as px
6 import streamlit as st
7
8 st.set_page_config(page_title="CD-SPAF", layout="wide")
9 st.title("CD-SPAF - Cloud-Drone Security Posture Assessment")
10 report = Path("outputs/report.json")
11 if not report.exists():
12     st.warning("Run 'python main.py --demo' first.")
13     st.stop()
14
15 data = json.loads(report.read_text(encoding="utf-8"))
16 findings = pd.DataFrame(data["findings"])
17 st.metric("Total Findings", len(findings))
```

dashboard/app.py

python main.py
--demo

outputs/
report.json

streamlit
dashboard

Execution and presentation are separate: one report can be re-examined without re-running the analysis.

Experimental Results

36

findings in the demonstrated run

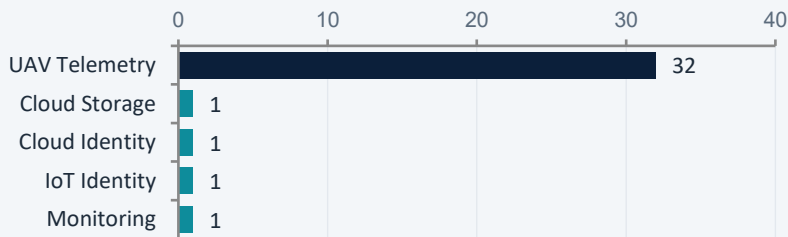
34

High severity (94.4%)

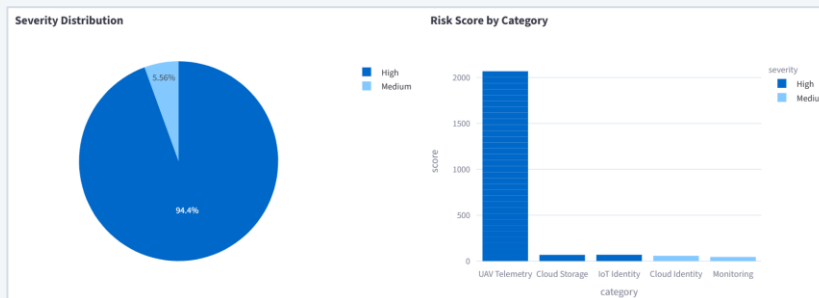
2

Medium severity (5.56%)

Findings by category (n = 36)



- 32 telemetry findings: 10 Isolation Forest, 20 DBSCAN, 2 threshold; one finding each for storage, identity, device identity and monitoring
- All 36 findings received a recommendation; every telemetry finding scored 64.6
- The telemetry data was deliberately anomaly-heavy, so this is not a real-world risk rate



TEL-THR-275	Telemetry anomaly detected (Threshold)	High	UAV-001	UAV Telemetry
AWS-S3-001	Telemetry storage encryption required	High	demo-telemetry	Cloud Storage
AWS-IAM-001	IAM least privilege	Medium	demo-iam-role	Cloud Identity
UAV-CERT-001	Unique drone certificate	High	UAV-001	IoT Identity
AWS-LOG-001	Centralised logging enabled	Medium	demo-account	Monitoring

finding_id	NIST CSF 2.0	OWASP IoT	Remediation
TEL-ISO-19	DE.CM	Monitoring	Investigate anomalous telemetry, validate mission constraints, and improve telemetr
TEL-ISO-53	DE.CM	Monitoring	Investigate anomalous telemetry, validate mission constraints, and improve telemetr
TEL-ISO-68	DE.CM	Monitoring	Investigate anomalous telemetry, validate mission constraints, and improve telemetr
TEL-ISO-77	DE.CM	Monitoring	Investigate anomalous telemetry, validate mission constraints, and improve telemetr

Top to bottom: dashboard (Fig. 5.8), findings table, recommendations table

Discussion and Evaluation

Objectives revisited

1 Identify weaknesses	Achieved
2 Design on NIST CSF 2.0 and Zero Trust	Achieved at design level
3 Prototype with cloud and telemetry rules	Partly: cloud data simulated
4 Risk assessment by severity	Demonstrated
5 Evaluate and recommend	Demonstrated: all 36 mapped

What the results mean: The results show that UAV and cloud security findings can be combined into one structure and prioritised together. This demonstrates the technical feasibility of the framework, but it does not prove real-world risk reduction.

Strengths

- Project uses a modular Python design, each part testable.
- Combines one pipeline for UAV and cloud findings
- provides recommendations based on security categories.

Limitations

- project uses an engineered dataset, so the results may not represent real-world situations.
- No accuracy metrics were measured, and the risk formula was not validated.
- Multi-cloud shown at design level only

Challenges

- project scope had to be limited to fit an MSc project.
- Anomalies were simulated and were not real attacks.
- All telemetry findings received the same risk score of 64.6.

Conclusion and Future Work



Contributions and achievements

- CD-SPAF is a Python-based security assessment framework.
- It combines UAV telemetry, device identity and cloud security checks.
- The prototype identified 36 findings in five categories.
- The project demonstrated the feasibility of the framework.



Current limitations

- The project used simulated UAV telemetry.
- Cloud data was predefined; no live AWS or Azure connection was used.
- Multi-cloud integration was demonstrated at the design level.
- The risk model has not been validated with real-world data.



Realistic future work

- Integrate live AWS and Azure services.
- Test the framework with real-time UAV telemetry.
- Use a labelled dataset to measure detection accuracy.
- Validate the risk model using expert feedback or incident data.
- Expand the security policies and add continuous monitoring.

Key message: In conclusion, my project demonstrates that UAV and cloud security findings can be assessed in one framework. The prototype works with simulated data, but further testing is needed for real-world use.

Selected References

- [1] H. Shakhathreh et al., "Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges," IEEE Access, vol. 7, pp. 48572–48634, 2019, doi: 10.1109/ACCESS.2019.2909530.
- [2] D. He, S. Chan, and M. Guizani, "Communication security of unmanned aerial vehicles," IEEE Wireless Commun., vol. 24, no. 4, pp. 134–139, Aug. 2017, doi: 10.1109/MWC.2016.1600073WC.
- [3] National Institute of Standards and Technology, "The NIST Cybersecurity Framework (CSF) 2.0," NIST, Gaithersburg, MD, USA, NIST CSWP 29, Feb. 2024, doi: 10.6028/NIST.CSWP.29.
- [4] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero trust architecture," NIST, Gaithersburg, MD, USA, NIST SP 800-207, Aug. 2020, doi: 10.6028/NIST.SP.800-207.
- [5] Amazon Web Services, "Security pillar – AWS Well-Architected Framework," AWS Documentation. Accessed: Sep. 25, 2026. [Online]. Available: <https://docs.aws.amazon.com/wellarchitected/latest/security-pillar/welcome.html>
- [6] Microsoft, "Microsoft cloud security benchmark v1 overview," Microsoft Learn. Accessed: Sep. 25, 2026. [Online]. Available: <https://learn.microsoft.com/en-us/security/benchmark/azure/overview-mcsb-v1>
- [7] A. R. A. Rahumath Ali and P. Deepalakshmi, "Cloud security posture management: Automating risk detection, compliance enforcement, and vulnerability remediation in cloud infrastructure," in Proc. 1st Int. Conf. Sustainability Technol. Adv. Eng. Domain (SUSTAINED), Dec. 2024, pp. 1–6, doi: 10.1109/SUSTAINED63638.2024.11073903.
- [8] Joint Task Force Transformation Initiative, "Guide for conducting risk assessments," NIST, Gaithersburg, MD, USA, NIST SP 800-30 Rev. 1, Sep. 2012, doi: 10.6028/NIST.SP.800-30r1.
- [9] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in Proc. 8th IEEE Int. Conf. Data Mining (ICDM), Dec. 2008, pp. 413–422, doi: 10.1109/ICDM.2008.17.
- [10] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in Proc. 2nd Int. Conf. Knowl. Discovery Data Mining (KDD-96), 1996, pp. 226–231.

Selected from the dissertation reference list; numbering matches the dissertation.



Thank you

Questions?

Shahriar Badhan

MSc Cyber Security and Digital Forensics

UWTSD | Supervisor: Dr Dipo Dunsin