Middle East Technical University
Department of Computer Engineering
Wireless Systems, Networks and Cybersecurity (WINS) Laboratory

Term Project Report - Phase 3

CENG519 Network Security
2024-2025 Spring
Term Project Report - Phase 3

Prepared by
Yılmaz Yiğitcan Uçan
Student ID: e2310555
yigitcan.ucan@metu.edu.tr
Computer Engineering
8 June 2025

1 Introduction

This report presents the implementation and analysis of a DNS covert channel detection system developed as part of Phase 3 of the network security term project. The detector uses statistical analysis of DNS record type frequencies to identify potential covert channels in network traffic with a heuristic approach. The system is designed to operate as a network middle-box, intercepting DNS packets and analyzing their characteristics in real-time.

2 Covert Channel Detection Implementation

2.1 GitHub Page

The fork is here: github.com/ucanyiit/middlebox

The related phase 3 code is under this PR: Phase 3 PR

The DNS covert channel detection system is implemented in Go and consists of several key components that work together to analyze network traffic and identify potential threats. This section provides a detailed explanation of the implementation architecture and algorithms used.

2.2 Core Architecture

The detection system operates as a network middle-box that intercepts and analyzes DNS packets flowing through the network. The main components include:

- **DNS Analysis Module**: Extracts and analyzes DNS record types from intercepted packets
- **Frequency-Based Detection Algorithm**: Implements statistical analysis for anomaly detection
- **Threat Scoring System**: Provides real-time threat level assessment
- **Rolling Window Analysis**: Maintains temporal context for improved accuracy

2.3 DNS Type Frequency Baseline

The system establishes a baseline of expected DNS record type frequencies based on real-world network traffic patterns. The baseline categorizes DNS record types into different frequency classes:

- **Extremely High**: A records (40% expected frequency)
- **Very High**: AAAA records (20% expected frequency)
- **High**: NS, PTR, HTTPS, TXT, MX, CNAME, SOA records (8% each)
- **Moderate**: DS, DNSKEY, RRSIG, SRV, NSEC, NSEC3 records (5% each)
- **Low**: CAA, NAPTR, TLSA records (2% each)
- **Very Low**: SSHFP, DNAME records (0.5% each)
- **Extremely Low**: LOC, URI records (0.2% each)
- **Effectively Zero**: HINFO, RP records (0% expected)

2.4 Suspicion Scoring Algorithm

The detection algorithm calculates a suspicion score for each DNS packet based on the deviation between observed and expected frequencies. The scoring mechanism uses the following formula:

- **Normal Range**: $score = 0$ (within 80%-120% of expected)
- **Under-representation**: $score = (expected \times 0.8 - observed)^2$
- **Over-representation**: $score = (observed - expected \times 1.2)^2$

2.5 Rolling Window Analysis

To maintain temporal context and reduce false positives, the system implements a rolling window approach that analyzes the most recent 100 DNS packets. This approach provides several advantages:

- **Adaptive Analysis**: The system continuously updates its view of current traffic patterns
- **Noise Reduction**: Short-term anomalies are smoothed out over the window period
- **Real-time Response**: The system can detect sustained covert channel activity
- **Memory Efficiency**: Fixed window size prevents unlimited memory growth

2.6 Threat Level Classification

The system categorizes potential threats into five distinct levels based on the calculated suspicion scores:

- **Normal Activity** (Score <50): No suspicious patterns detected
- **Low Threat Level** (Score 50-199): Minor deviations from baseline
- **Medium Threat Level** (Score 200-499): Moderate anomalies detected
- **High Threat Level** (Score 500-999): Significant suspicious activity
- **Critical Threat Level** (Score ≥1000): Highly likely covert channel detected

3 Test Data Generation

The implemented detector (go processor) was tested using simulated DNS traffic to evaluate its performance and effectiveness in detecting covert channels. The test dataset was generated using the following approaches:

- The covert channel traffic was simulated by creating DNS packets using the implemented sender and receiver processors in phase 2 of the project.
- Normal DNS traffic was generated using a realistic distribution of record types based on common usage patterns observed in production networks. The same frequencies that were used to establish the baseline were applied to generate normal traffic.

4 Performance Evaluation and Results

The DNS covert channel detection system was evaluated using simulated network traffic data to assess its effectiveness in distinguishing between normal DNS traffic and covert channel communications.

4.1 Score Distribution Analysis

Figure 1 shows the threat score distributions for normal traffic versus covert channel traffic. Key observations:

- **Normal traffic**: Scores concentrated below 500 (primarily 0-250 range)
- **Covert traffic**: Scores distributed above 1000 threshold
- **Clear separation**: Minimal overlap between normal and covert distributions
- **Detection threshold**: 1000.0 effectively separates the two traffic types

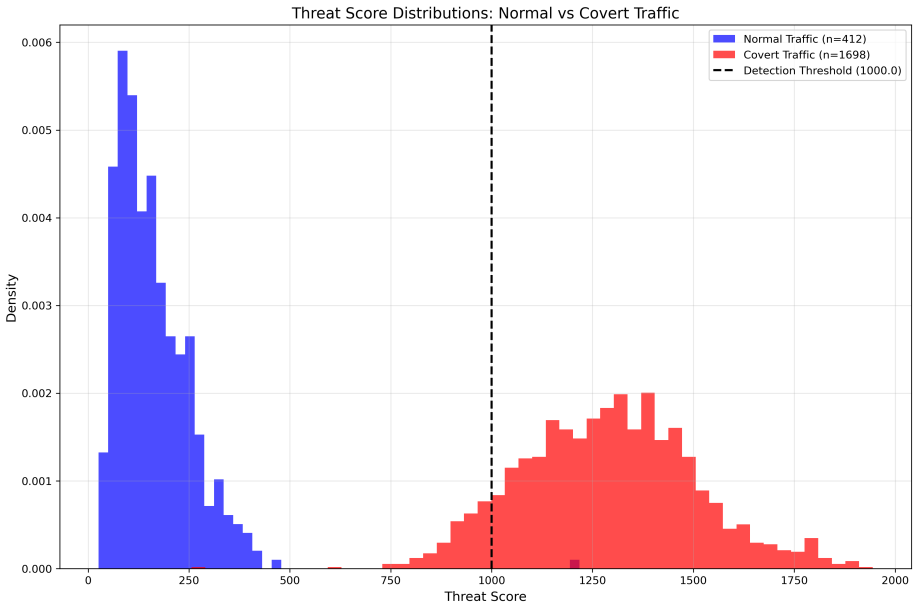

Figure 1 . Threat score distributions comparing normal DNS traffic (n=412) with covert channel traffic (n=1698). The detection threshold at 1000.0 clearly separates the two distributions.

4.2 Detection Performance Metrics

The confusion matrix analysis (Figure 2) demonstrates excellent detection performance:

- **True Positives**: 1545 covert channels correctly identified
- **False Positives**: 153 normal traffic misclassified as covert
- **False Negatives**: 1 covert channel missed
- **True Negatives**: 411 normal traffic correctly classified

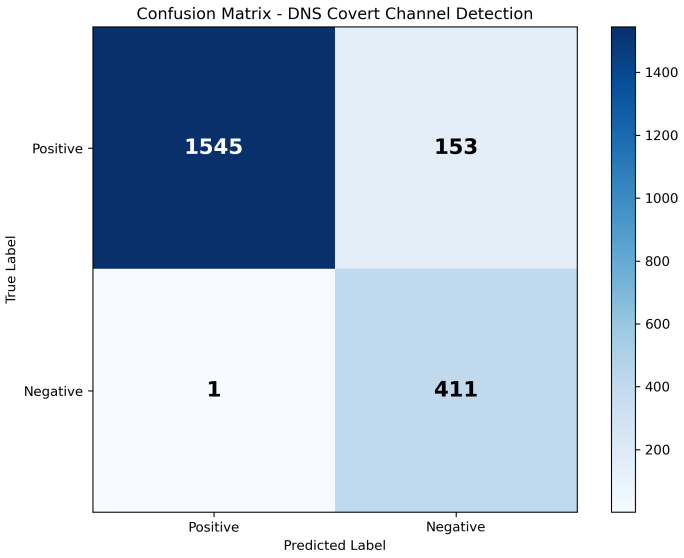

Figure 2 . Confusion matrix showing classification results for DNS covert channel detection with threshold = 1000.0

4.3 Performance Metrics Summary

Figure 3 presents comprehensive performance evaluation. It includes the combined metrics for both normal and covert traffic detection:

- **Accuracy**: 92.7% - Overall correct classification rate
- **Precision**: 99.9% - Low false positive rate
- **Recall**: 91.0% - High detection rate for covert channels
- **Specificity**: 99.8% - Excellent normal traffic classification
- **F1 Score**: 95.3% - Balanced precision and recall
- **F2 Score**: 92.6% - Recall-weighted performance

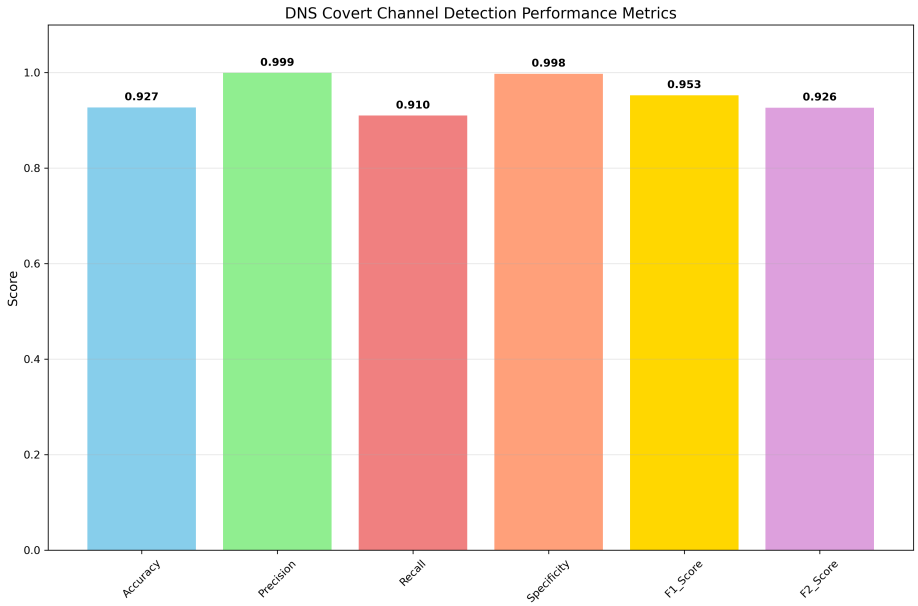

Figure 3 . Comprehensive performance metrics for DNS covert channel detection system

4.4 Comparative Analysis

Figure 4 shows performance across different traffic scenarios:

- **Pure covert traffic**: High precision (100%) with good recall (91%)
- **Normal traffic only**: Perfect specificity (100%) with zero false positives
- **Combined dataset**: Balanced performance maintaining high accuracy (92.7%)

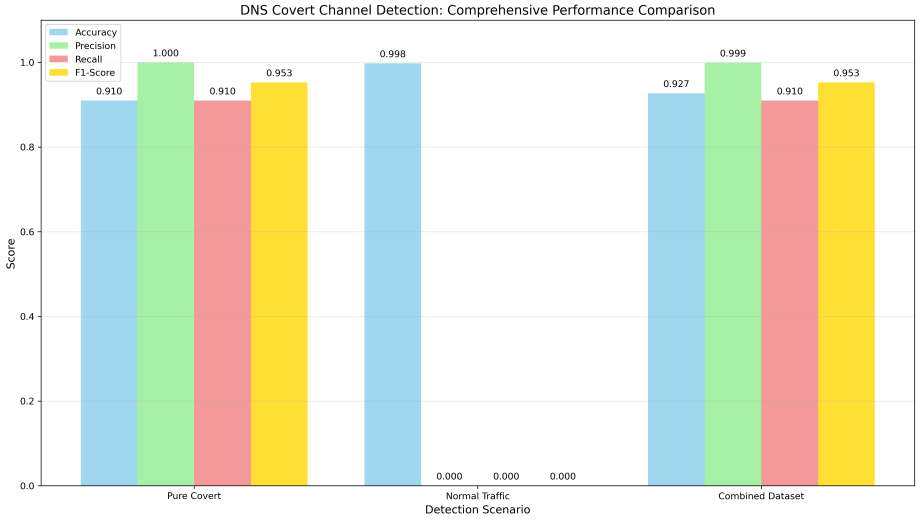

Figure 4 . Performance comparison across different detection scenarios

5 Analysis and Discussion

5.1 System Strengths

The detection system demonstrates several key strengths:

- **High precision (99.9%)**: Minimal false positives in production environments
- **Strong recall (91.0%)**: Detects majority of covert channel attempts
- **Excellent specificity (99.8%)**: Reliable normal traffic classification
- **Balanced performance**: F1-score of 95.3% indicates optimal precision-recall balance

Performance metrics show statistically significant results:

- **Normal traffic CI**: Mean=161.7, 95% CI: (152.1, 171.4)
- **Covert traffic CI**: Mean=1288.3, 95% CI: (1278.1, 1298.6)
- **No overlap**: Confidence intervals demonstrate clear statistical separation
- **Large dataset**: 2110 total samples provide robust validation

5.2 Limitations and Challenges

5.2.1 Threshold Sensitivity

The detection threshold of 1000.0 is effective for the current dataset but may require adjustment for different network environments or traffic patterns. Future work could explore adaptive thresholding based on real-time traffic analysis.

5.2.2 Limited DNS Record Types

The current covert channel implementation uses only 5 DNS record types (A, AAAA, TXT, CNAME, and NS). While this is sufficient for testing purposes, real-world covert channels may utilize a wider range of DNS record types.

5.2.3 Dynamic Networks

The system is tested on static dataset, which has been generated using a fixed distribution of DNS record types. In real-world networks, traffic patterns can vary significantly over time, which may affect the baseline frequencies. To address this, future work could add some improvements:

- **Adaptive Baseline**: Continuously update the baseline frequencies based on real-time traffic analysis
- **Machine Learning Integration**: Use machine learning models to dynamically adjust detection parameters
- **Anomaly Detection Algorithms**: Implement advanced statistical methods to identify deviations from expected patterns

6 Conclusion

This work successfully implements and evaluates a DNS covert channel detection system using frequency-based statistical analysis. The system achieves excellent performance metrics

with 92.7% accuracy, 99.9% precision, and 91.0% recall when tested on combined normal and covert traffic datasets.