**Project Proposal**

**Detecting Phishing Websites with Machine Learning**

**Team Members:**

1. – Hadeer Amr Fawzy (22010450) cybersecurity
2. – Farida Ahmed (2206160) cybersecurity
3. – Muhammed Salah (22010448) cybersecurity
4. –Marwan Gaber Ramdan (2206167) cybersecurity
5. – Omar Ibrahim Ahmed (2206209) cybersecurity
6. – Youssef Tamer Muhammed Ali (2206208) cybersecurity
7. – Ahmed Saber Ahmed (2203185) ai

**Introduction**

Phishing attacks trick people into giving away sensitive information by pretending to be trustworthy websites. Hackers keep improving their tactics, making it harder for traditional security tools to keep up.

**Our Goal**We will build a machine learning (ML) model that can automatically detect phishing websites by analyzing their URLs and other features. This will help improve cybersecurity defenses against these scams.

**Why This Matters**

Phishing causes billions in losses every year and current methods like blacklists are slow and outdated also ML can learn patterns and detect new phishing tricks faster.

**Literature Review (What Others Have Done)**

We looked at recent research to understand how others have tackled this problem:  
Recent studies show machine learning is highly effective for phishing detection  
**Phishing Website Detection using ML Algorithms 2018**  
compared 7 algorithms including RF, SVM, and Neural Networks  
random Forest achieved highest accuracy (97.3%) using URL and HTML features  
found that hybrid features work better than URL-only analysis  
**Phishing Detection Using ML Model Development 2024**  
developed a real-time detection system using XGBoost  
added new features like "domain registration age"  
achieved 98.1% accuracy while reducing false positives  
highlighted importance of feature selection  
**Comparison of SVM and RF for Web Security 2023**  
Tested on 15,000 website samples  
Random Forest performed better 96.2% vs SVM's 94.7%  
Noted RF handles imbalanced data better  
SVM trained faster but was less accurate

Key Insights Random Forest consistently performs well in phishing detection, combining multiple feature types improves results and most studies don't test on very recent phishing samples  
Our Improvement We'll address the gap by testing on newly reported phishing sites ,combining the best features from these studies and comparing both accuracy and detection speed

**Dataset to Be Used**

We’ll check which will be more helpful for us

1. phishing dataset from UCI https://archive.ics.uci.edu/dataset/327/phishing+websites

(11,000+ websites, labeled as "phishing" or "safe").

1. [Mendeley Phishing Dataset](https://data.mendeley.com/datasets/h3cgnj8hft/1) (adds more features like JavaScript tricks).
2. PhishStorm https://research.aalto.fi/en/datasets/phishstorm-phishing-legitimate-url-dataset
3. OpenPhish live feed for testing <https://www.openphish.com/>

Why These because it is ready-to-use no need to collect data manually and balanced similar number of phishing and safe websites.

**Proposed Methodology & Approach**

Step 1: Data Preprocessing Clean the data: Remove duplicates, fix missing values. Extract key features as URL length , Use of HTTPS ,Number of suspicious symbols (@, -, .com) and Domain age (phishing sites are often new).

Step 2: Model Building We’ll test 3 machine learning models Random Forest (works well with many features) Support Vector Machine (SVM) (good for binary classification) Logistic Regression (simple baseline for comparison).

Step 3: Evaluation metrics Accuracy, Precision, Recall, F1-Score test against real-world phishing URLs from [PhishTank](https://www.phishtank.com/" \t "_blank).

**Type of Attack We’ll Detect**

URL Spoofing as paypa1.com instead of paypal.com  
Fake Login Pages stealing passwords

**Expected Results & Evaluation**

What We ExpectHigh accuracy (~95%) in detecting phishing sites random forest should perform best (based on past research) SVM will be faster but slightly less accurate.

**Key Performance Indicators (KPIs)**Accuracy >90%  
Low false-positive rate <5%  
Training Time < 1 min

**Final Thoughts**  
This project is practical solves a real cybersecurity problem. If our model works well, it could be turned into a browser plugin to warn users about phishing sites in real time.