AI IN SOCIAL ENGINEERING AND PHISHING CAMPAIGNS

BEHAVIOR-BASED URL DETECTION TOOL

PRESENTED BY: PRANAV, ATUL & ZAID

INTERNSHIP: DIGISURAKSHA CYBERSECURITY INTERNSHIP 2025

WHAT IS SOCIAL ENGINEERING?

- Definition: A manipulation technique that exploits human error to gain private information, access, or valuables.
- Types:
  - Phishing – fake emails/websites
  - Baiting – luring with free offers/media
  - Pretexting – posing as authority (e.g., IT support)
  - Tailgating – physically following someone into restricted areas
- Goal: Trick users into making security mistakes.

WHAT IS PHISHING?

- A type of social engineering attack that tricks users into revealing sensitive data (e.g., passwords, bank details).
- Delivery methods:
  - Emails (most common)
  - Fake websites
  - Text messages (smishing)
  - Phone calls (vishing)
- Example: "Your account has been locked. Click here to verify."

HOW AI IS USED IN CYBERSECURITY

- Anomaly detection: Detects behavior that deviates from the norm.
- Spam/phishing filters: ML algorithms flag suspicious emails.
- Threat intelligence: AI scans forums/dark web for leaked data.
- Automation: Speeds up incident response with AI bots.

HOW AI IS MISUSED BY ATTACKERS

- Natural Language Generation (NLG):
  - AI writes personalized, grammatically correct phishing emails.
- Voice cloning:
  - Mimics real people (e.g., CEOs) using deep learning.
- Automated target profiling:
  - Scrapes public data from social media to craft tailored attacks.
- Chatbots:
  - Fake customer support bots lure victims into giving credentials.

RISE OF AI-DRIVEN PHISHING ATTACKS

- Growth: Over 500 million phishing attacks in 2023.
- AI accelerates phishing:
  - Faster, cheaper, more personalized.
  - Harder to detect because emails appear more “human”.
- Low barrier: Attackers use AI without deep technical skills.

LIMITATIONS OF TRADITIONAL SECURITY

- Keyword filters fail when AI-generated content avoids spam triggers.
- Static blacklists can’t keep up with fast-changing domains.
- User training is often forgotten or ignored.
- Legacy antivirus tools lack behavioral analysis features.

CONSEQUENCES OF AI-ENHANCED PHISHING

- Financial loss: $10.3 billion in phishing losses (FBI IC3, 2022)
- Reputation damage: Customer trust collapses after data breaches.
- Data theft: Access to corporate systems and IP leaks.
- Nation-state risk: Attacks on government and infrastructure.

USING AI TO COUNTER PHISHING

- Email anomaly detection:
  - Scans sender behavior, writing style, timing patterns.
- Phishing simulators:
  - AI generates fake attacks for training purposes.
- Neural networks:
  - Classify emails and websites based on visual or structural features.

PHISHING URL AND DOMAIN ANALYSIS WITH AI

- Key indicators AI analyses:
  - Misspelled domains (e.g., "go0gle.com")
  - HTTPS usage and certificate info
  - Number of redirects or hidden links
- Techniques:
  - Logistic regression, Random Forest, or deep learning (CNNs) for URL classification
- Data Sources:
- PhishTank, Kaggle datasets

DEEPFAKE VOICE PHISHING ATTACK

- Case: UK energy firm tricked into sending €220,000.
- How: Voice-cloning software mimicked CEO's voice.
- Impact: Money transferred to attacker's account in Hungary.
- Insight: Audio AI makes vishing more believable and dangerous.

WHAT'S THE TOOL ABOUT?

- This is a machine learning–based URL classification tool designed to detect and categorize suspicious or harmful web links by analyzing their structure.
- It uses feature extraction techniques to convert a URL into numerical data, then feeds it into a trained model to classify the URL into one of four categories:
  - ✓ Benign – Safe to visit
  - ⚠ Phishing – Fake websites trying to steal credentials
  - ⊘ Defacement – Websites whose content has been maliciously altered
  - ⯃ Malware – Sites that try to install harmful software

WHY WAS THIS TOOL MADE?

- Cyber threats are growing
  - Every day, users unknowingly click on dangerous links leading to phishing or malware. This tool helps prevent attacks before they happen.
- It's practical and impactful
  - The tool can be used in the real world — in email filters, browsers, or cybersecurity apps — making it more than just a theory project.
- Applies machine learning to real problems
  - This project allowed me to apply ML practically in cybersecurity — feature extraction, model training, and prediction, all in one pipeline.
- Simple, fast, and effective
  - It works quickly with just a URL input, needs no large system, and provides accurate results using a lightweight Random Forest model.
- Educational and expandable
  - It’s a great learning base for beginners and can be extended further — with URL reputation APIs, deep learning, or threat intelligence databases.

WHO CAN USE IT & WHERE IT CAN BE USED?

- Individual users
  - **General internet users** can paste a suspicious URL and check if it's safe before clicking.
- Students and learners
  - **Cybersecurity students** can study how ML models detect malicious URLs.
- Analysts
  - Quickly classify suspicious links found in phishing emails, logs, or social engineering attempts.
- Developers & engineers
  - Can integrate it into **Web applications** (to validate user-submitted URLs), APIs or bots that interact with third party links.
- Educators and trainers
  - Can use this tool to **demonstrate phishing techniques** and **train users** on how to identify harmful URLs.

HOW DOES IT WORK?

- File 1: extract_features.py
- Converts a URL into numerical features like:
  - URL length
  - Use of https
  - Number of subdomains
  - Suspicious words (like “login”, “bank”)

- File 2: train_model.py
- Trains a Random Forest model using a dataset of labelled URLs.
- Extracts features using extract_features.py.
- Splits the data, trains the model, evaluates accuracy.
- Saves the trained model as model.pkl.

- File 3: check_url.py
- Loads the saved model.
- Takes a new URL input.
- Extracts features using extract_features.py.
- Predicts if the URL is:
  - Benign
  - Phishing
  - Defacement
  - Malware

CODE/TOOL IMPLEMENTATION

- Developed using Python + Scikit-learn (No LINUX dependencies)
- Trained classifier to label URLs into 4 behavior types
- Exported as REST API and GUI CLI app
- Lightweight enough for real-time detection

```
Class distribution:
type
0    428103
2     96457
1     94111
3     32520
Name: count, dtype: int64

Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.92      0.95     85778
           1       0.69      0.85      0.76     18836
           2       0.96      0.98      0.97     19104
           3       0.97      0.93      0.95      6521

    accuracy                           0.92    130239
   macro avg       0.90      0.92      0.91    130239
weighted avg       0.93      0.92      0.92    130239


Model saved as model.pkl
```

FEATURE ENGINEERING INSIGHTS

- Extracted 30+ features from URLs, e.g.:
  - 🔗 Length, . count, special characters
  - ☐ Entropy of domain
  - 🌐 Use of suspicious keywords (e.g., “login”, “update”)
- Recursive Feature Elimination (RFE) used to select top features
- Top 5 impactful features shown to have >80% predictive contribution

CODE/TOOL DEMONSTRATION

- Input: train_model.py (Generates report)
- Input: User enters or pastes a URL
- Output: Classification result with probability scores
- Optional logging and alert system

```
C:\Users\Pranav\OneDrive\Tài liệu\Cyber Security\Final project\Threat-URL-Detector\tool\source_code>python check_url.py
Enter a URL to check: br-icloud.com.br
Result: Phishing

C:\Users\Pranav\OneDrive\Tài liệu\Cyber Security\Final project\Threat-URL-Detector\tool\source_code>python check_url.py
Enter a URL to check: mp3raid.com/music/krizz_kaliko.html
Result: Benign

C:\Users\Pranav\OneDrive\Tài liệu\Cyber Security\Final project\Threat-URL-Detector\tool\source_code>python check_url.py
Enter a URL to check: http://www.garage-pirenne.be/index.php?option=com_content&view=article&id=70&vsig70_0=15
Result: Defacement

C:\Users\Pranav\OneDrive\Tài liệu\Cyber Security\Final project\Threat-URL-Detector\tool\source_code>python check_url.py
Enter a URL to check: http://www.824555.com/app/member/SportOption.php?uid=guest&langx=gb
Result: Malware
```

FUTURE ENHANCEMENTS

- Visual Spoof Detection
  - Compare page screenshots to detect fake brand logos.
- User Feedback Learning
  - Improve accuracy using reinforcement from user input.
- Browser Extension Integration
  - Detect phishing attempts in real-time while browsing.
- Threat Intelligence APIs
  - Enrich URL analysis with data from OpenPhish, VirusTotal, etc.

THANK YOU

- Project: AI in Phishing & URL Behavior Detection
- GitHub: https://github.com/prannaw/Pranav_Shivaji_Dhumale
- YouTube demo: https://youtu.be/SzvPzcSCwio