

PhishNet: Phishing Awareness Analysis System

By: Devin Giles

Carlos Rey

Samuel Castellanos

Ahsan Sanders

CIS 4951: Capstone II

Prof. Masoud Sadjadi

4/15/2026

Problem Definition

- Phishing remains a major cybersecurity threat driven by human error.
- Users often cannot distinguish legitimate vs phishing emails
- Organizations struggle to measure user awareness effectively
- **Problem:**
There is a lack of clear tools to evaluate and visualize phishing susceptibility.

System Architecture

- PhishNet is designed as a modular system with separate components:
- **Data Layer:** Simulated and research-based phishing datasets
- **Backend (FastAPI):** Processes data and calculates metrics
- **Frontend Dashboard:** Displays results visually
- **Evaluation Layer:** Interprets results and trends
- **Data Flow:**
Data → Backend API → Dashboard → User Analysis

Implementation

Our system is built using a connected backend and frontend architecture:

- **Backend:**
 - Built with FastAPI
 - Processes simulated phishing data
 - Computes key metrics:
 - Total users
 - Failure rate
 - Reporting rate
 - Exposes results through an API endpoint
- **Frontend:**
 - Built with HTML, CSS, and JavaScript
 - Uses Chart.js for data visualization
 - Fetches data from backend API in real-time
 - Displays interactive charts and filtering options

Demonstration overview

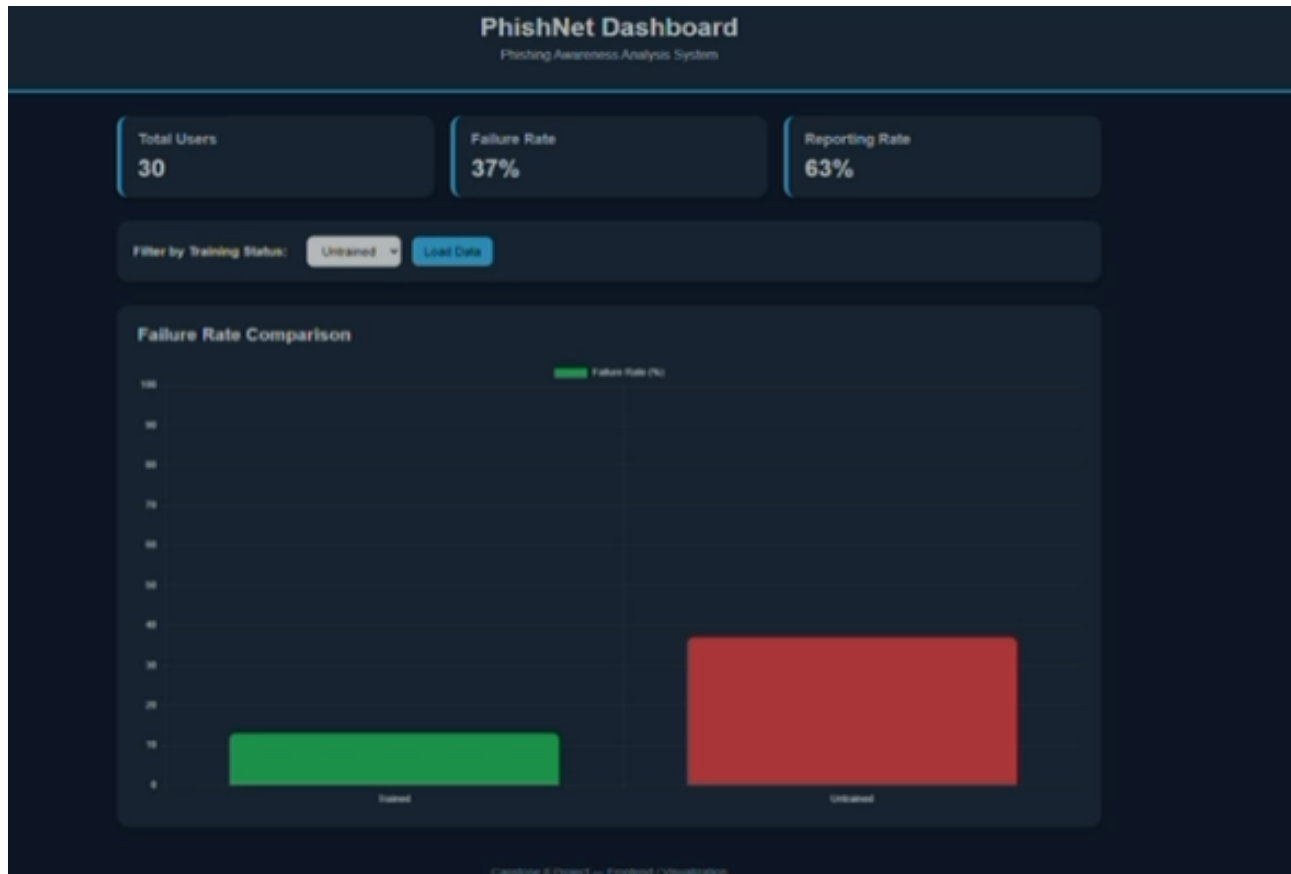


Image description

The dashboard visualizes phishing awareness using a dataset of 30 users. The backend processes user actions to calculate failure and reporting rates, while the frontend retrieves this data and displays it in summary cards and a comparison chart. The filter isolates trained and untrained users, highlighting performance differences between the two groups.

The dashboard provides a real-time view of phishing performance:

- Displays key metrics:
 - Total users
 - Failure rate
 - Reporting rate
- Includes filtering:
 - All users
 - Trained users
 - Untrained users
- Visual comparison using charts

Testing & Evaluation Results

- The system was evaluated using a behavior-based phishing dataset.
- The dataset includes simulated users with recorded actions:
 - link clicks (failure)
 - phishing reports (success)
- **Evaluation Approach:**
 - Metrics are calculated dynamically from user behavior
 - Data can be filtered by training status (trained vs untrained)
- **Key Findings:**
 - Trained users show lower failure rates and higher reporting rates
 - Untrained users show higher failure rates and lower reporting rates
- The dashboard allows direct comparison between both groups
- **Conclusion:**

The results demonstrate that phishing awareness training significantly improves user performance.

Challenges & Lessons Learned

- **Challenges:**

- Connecting frontend to backend API
- Managing consistent data structure
- Setting up development environments

- **Lessons Learned:**

- Clear system architecture is critical
- Data visualization improves understanding
- Team collaboration is essential for integration

Future work

- **Content:**

- Add real email simulation system
- Expand dataset with more users
- Improve dashboard visuals and analytics
- Deploy system online (cloud-based)
- Add user authentication

Why :

These improvements would make the system more realistic, scalable, and applicable to real-world cybersecurity environments, allowing for deeper analysis and broader use.

Conclusion

- PhishNet demonstrates an effective approach to analyzing phishing awareness.
 - Phishing is largely a human-focused problem
 - Training reduces user susceptibility
 - Visualization helps organizations understand risk
- **Final Insight:**
Combining simulation, backend processing, and visualization provides meaningful security insights.