VIETNAM GENERAL CONFEDERATION OF LABOUR

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**PHẠM NGỌC LINH - 521H0360**

**LÂM NGUYỄN ANH THY - 521H0377**

**NGUYỄN ĐÌNH VIỆT HOÀNG - 522H0120**

**DETECT PHISHING WEBSITES USING MACHINE LEARNING**

**FINAL REPORT**

**INTRODUCTION TO INFORMATION SECURITY**

**HO CHI MINH CITY, YEAR 2024**

VIETNAM GENERAL CONFEDERATION OF LABOUR

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**PHẠM NGỌC LINH - 521H0360**

**LÂM NGUYỄN ANH THY - 521H0377**

**NGUYỄN ĐÌNH VIỆT HOÀNG - 522H0120**

**DETECT PHISHING WEBSITES USING MACHINE LEARNING**

**FINAL REPORT**

**INTRODUCTION TO INFORMATION SECURITY**

Instructor

**PhD. Huỳnh Ngọc Tú**

**HO CHI MINH CITY, YEAR 2024**

**ACKNOWLEDGEMENT**

We sincerely thank PhD. Huỳnh Ngọc Tú for teaching us the Introduction to Information Security course with great enthusiasm. We want to express our deep appreciation for the dedication and professional knowledge that you shared with us. Through your classes, we gained a better understanding of the fundamental aspects of the Introduction to Information Security, thanks to your detailed explanations and practical applications. You helped us grasp the knowledge and apply it effectively. Finally, we extend our heartfelt gratitude to PhD. Huỳnh Ngọc Tú for your commitment and invaluable support throughout our learning journey in this course. The skills and knowledge we acquired will continue to impact our future development. We sincerely thank you and wish your health, success, and happiness.

*Ho Chi Minh City, May 23, 2024*

*Authors:*

*Phạm Ngọc Linh*

*Lâm Nguyễn Ạnh Thy*

*Nguyễn Đình Việt Hoàng*

**DECLARATION OF AUTHORSHIP**

Our group assures that this is our own report and was guided by PhD. Huỳnh Ngọc Tú. The research content and results in this report are honest and have not been published in any form before. The figures in the tables used for analysis, comments, and evaluations were collected by the authors from various sources clearly stated in the reference section.

Additionally, the report includes some comments, evaluations, and data from other authors and organizations, all of which are cited and noted for their origin.

**If any fraud is detected, we fully take responsibility for the content of our midterm report for the second semester of the 2023-2024 academic year.** Ton Duc Thang University is not involved in any copyright or intellectual property violations that we may cause during the process (if any).

*Ho Chi Minh City, May 23, 2024*

*Authors:*

*Phạm Ngọc Linh*

*Lâm Nguyễn Ạnh Thy*

*Nguyễn Đình Việt Hoàng*

**INSTRUCTOR RUBRIC**

Supervisor’s Name: ……………………………………………………………..............

Comments: ……………………………………………………………………………...

Total Score Based on Rubric Evaluation: ………………………………………………

*Ho Chi Minh City, date … month … year …*

*Supervisor*

*(sign and write your full name**)*

Our team snippet uses a dataset of URLs called 'urldata.csv'. This is a dataset of URLs, with a 'Label' column indicating whether the URL is a phishing URL.

* Step 1: Load the Data

The dataset is loaded from a CSV file named urldata.csv using Pandas:

data = pd.read\_csv('urldata.csv')

This dataset contains information about various URLs, including features extracted from the URLs and a target variable indicating whether the URL is phishing (1) or legitimate (0). The column names likely include 'Domain' (the URL itself) and various numerical or categorical features describing the URL.

* Step 2: Encode the Target Variable

The target variable, 'Label', is encoded using LabelEncoder to convert the categorical labels into numerical values:

label\_encoder = LabelEncoder()

data['Label'] = label\_encoder.fit\_transform(data['Label'])

* Step 3: Split the Data into Features (X) and Target Variable (y)

The dataset is split into features (X) and target variable (y). The 'Domain' and 'Label' columns are excluded from the features set:

X = data.drop(['Domain', 'Label'], axis=1)

y = data['Label']

The data is further split into training and testing sets using train\_test\_split to allow for model validation:

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

* Step 4: Train the Random Forest Model

A RandomForestClassifier is initialized and trained on the training data:

rf\_classifier = RandomForestClassifier(n\_estimators=100, random\_state=42)

rf\_classifier.fit(X\_train, y\_train)

* Step 5: Model Evaluation

The trained model is evaluated on the testing set, and performance metrics such as accuracy and classification report are printed:

y\_pred = rf\_classifier.predict(X\_test)

print("Model Accuracy:", accuracy\_score(y\_test, y\_pred))

print("Classification Report:")

print(classification\_report(y\_test, y\_pred))

Feature importance is calculated and displayed to understand which features contribute most to the model's predictions:

feature\_importance = pd.DataFrame(rf\_classifier.feature\_importances\_, index=X.columns, columns=['importance']).sort\_values('importance', ascending=False)

print("Feature Importance:")

print(feature\_importance)

* Step 6: Function to Predict Domain Legality

A function predict\_domain\_legality is defined to check if a given domain exists in the dataset and predict its legality using the trained model:

def predict\_domain\_legality(domain):

if domain in data['Domain'].values:

domain\_row = data[data['Domain'] == domain].drop(['Domain', 'Label'], axis=1)

prediction = rf\_classifier.predict(domain\_row)

if prediction[0] == 1:

print(f"The domain '{domain}' is likely a phishing domain.")

else:

print(f"The domain '{domain}' is likely a legitimate domain.")

else:

print(f"The domain '{domain}' does not exist in the dataset.")

* Step 7: User Interaction

The user is prompted to input a domain to check its legality:

domain\_to\_check = input("Enter a domain to check its legality: ")

predict\_domain\_legality(domain\_to\_check)



