

Infosys Springboard 6.0

Python Task

1. Introduction to Python

Python is a versatile programming language used widely in data science, artificial intelligence, and cybersecurity analytics. In this task, Python is applied to analyze, preprocess, and predict cyber threats using machine learning and deep learning techniques.

2. Data Preprocessing and Analysis

1. Import necessary libraries:

```
import pandas as pd
import numpy as np
```

2. Load dataset:

```
data = pd.read_csv('CICIDS2017.csv')
```

3. Handle missing values:

```
data.fillna(0, inplace=True)
```

4. Normalize numerical features using MinMaxScaler for model compatibility.

3. Feature Selection and Model Building

1. Feature Selection:

```
from sklearn.feature_selection import SelectKBest, chi2
X_new = SelectKBest(chi2, k=10).fit_transform(X, y)
```

2. Machine Learning Model:

```
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier()
model.fit(X_train, y_train)
```

3. Deep Learning Model:

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
model = Sequential([Dense(64, activation='relu'), Dense(1, activation='sigmoid')])
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
```

4. Model Evaluation

1. Evaluate model performance:

```
from sklearn.metrics import classification_report, confusion_matrix
y_pred = model.predict(X_test)
print(classification_report(y_test, y_pred))
```

2. Visualize confusion matrix using Matplotlib or Seaborn for better insights.

5. Real-Time Threat Prediction

1. Simulate live network data using packet capture (PCAP) or Zeek logs.
2. Use trained model to classify traffic as benign or malicious in real-time.
3. Implement alert system using Streamlit or Flask for dashboard visualization.

6. Conclusion

Python enables robust cybersecurity analytics through its libraries and AI integration capabilities. It forms the core for data-driven threat detection systems and real-time AI model deployment.