Towards Effective Machine Learning Models for Ransomware Detection via Low-Level Hardware Information

Chutitep Woralert, Chen Liu, Zander Blasingame
Clarkson University
Potsdam, New York, U.S.A.

https://camel.clarkson.edu/

HASP 2024

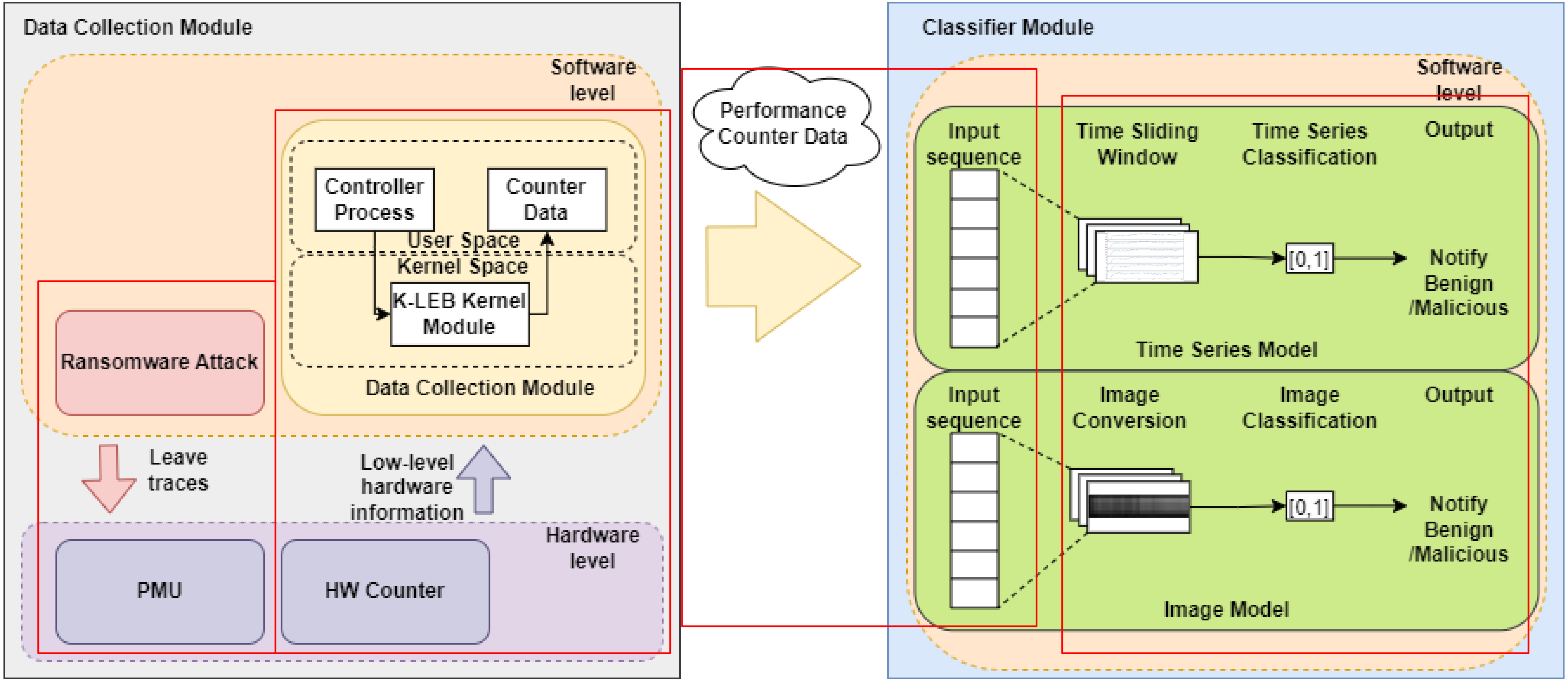
Motivation

- Reported 317.6M ransomware attack in 2023[1]
- Many techniques has been developed to fight ransomware
- Neural networks have gained popularity as a detection classifier
- Explore several state of the art models performance in detecting ransomware using low-level hardware information

Screenshot of Wannacry Ransomware Attack

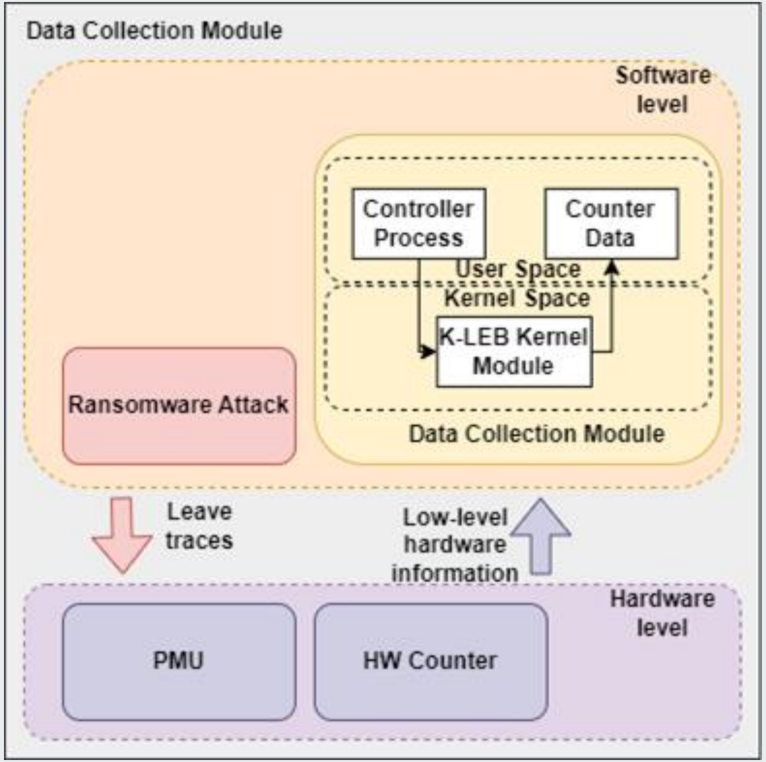
[1] SonicWall 2024 cyber threat report

Detection Framework[1]



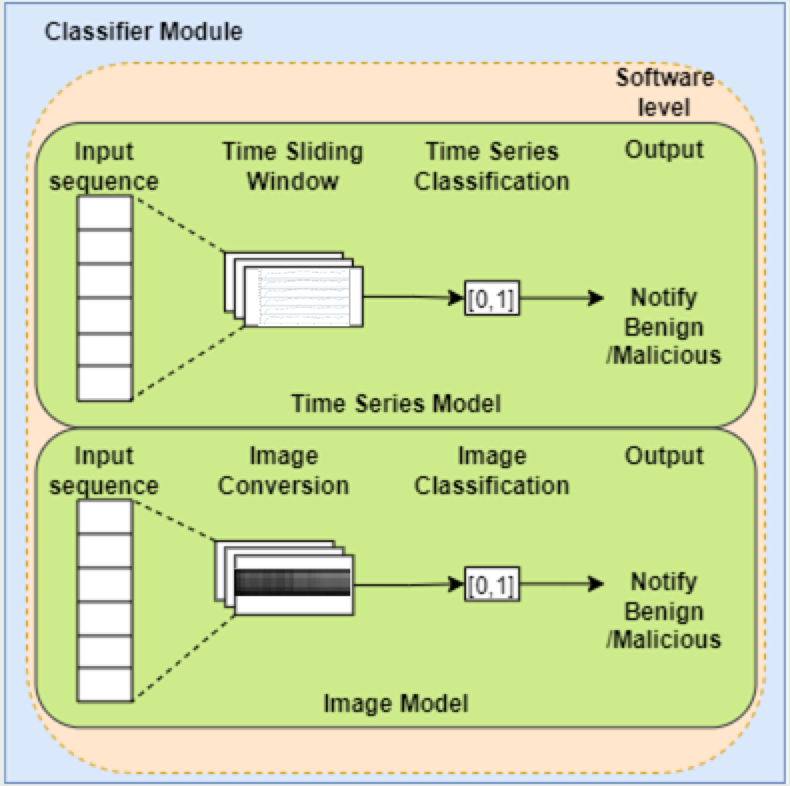
[1] C. Woralert, C. Liu and Z. Blasingame, "HARD-Lite: A Lightweight Hardware Anomaly Realtime Detection Framework Targeting Ransomware," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 70, no. 12, pp. 5036-5047, Dec. 2023, doi: 10.1109/TCSI.2023.3299532

Data Collection Module



- Collect low-level hardware information from the user machine
- Collect system wide hardware events
- The data is collected periodically as time series format

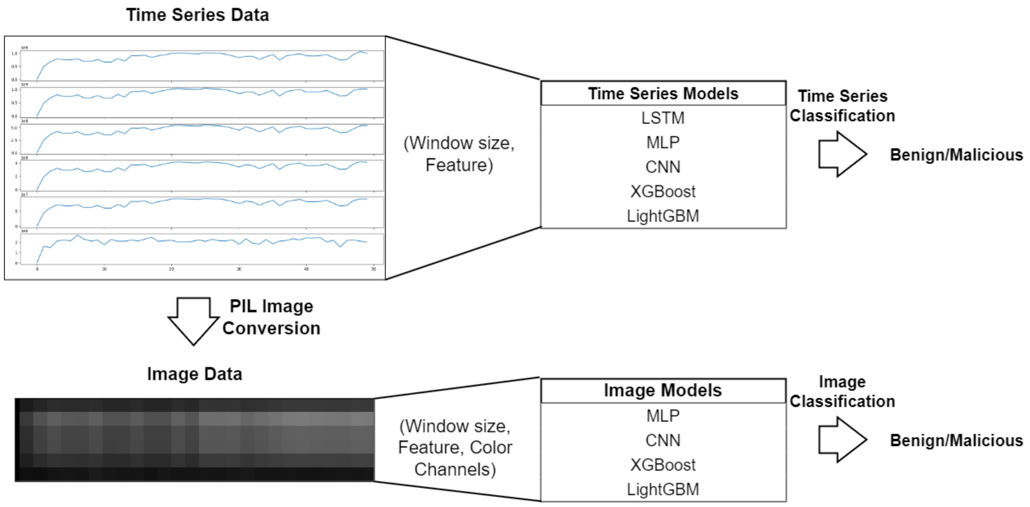
Classifier Module



- Classification models
  - Time series model
  - Image model
- Input features: 6 Hardware events:
  - Branch retire
  - Instruction retire
  - Data cache access
  - Load
  - Store
  - Last level cache miss
- Perform online analysis on the data from the data collection module

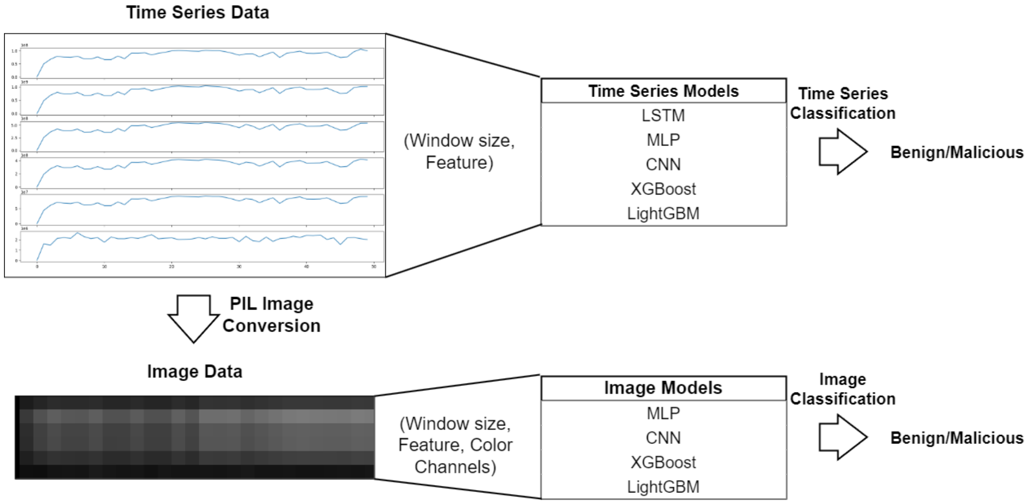
Neural Network Models

- Long-short term Memory (LSTM)
- Convolutional Neural Networks (CNNs)
- The Multilayer Perceptron (MLP)



Gradient Boosting Models

- eXtreme Gradient Boosting (XGBoost)
- Light Gradient Boosting Machine (LightGBM)



Experiment Setup

- Perform experiment on user machine with regular workloads
- Deploy data collection module to collects hardware events on the user machine
- Deploy classification module that processes the information received from the user machine
- Deploy ransomware attack on the user machine then monitor the classification result

Classification Results (Window size = 50)

Model/Dataset	Time Series Data				Image Data			
Window size 50	Accuracy	Precision	Recall	F1 Score	Accuracy	Precision	Recall	F1 Score
LSTM	97.05	98.77	95.27	96.99	N/A	N/A	N/A	N/A
XGBoost	99.81	99.93	99.70	99.81	99.73	99.93	99.53	99.73
LightGBM	99.77	99.89	99.65	99.77	99.78	99.95	99.61	99.78
MLP	98.41	99.07	99.07	98.73	99.40	99.31	99.49	99.40
CNN	97.94	97.43	98.56	97.97	99.91	99.93	99.89	99.91

Classification Results (Window size = 1000)

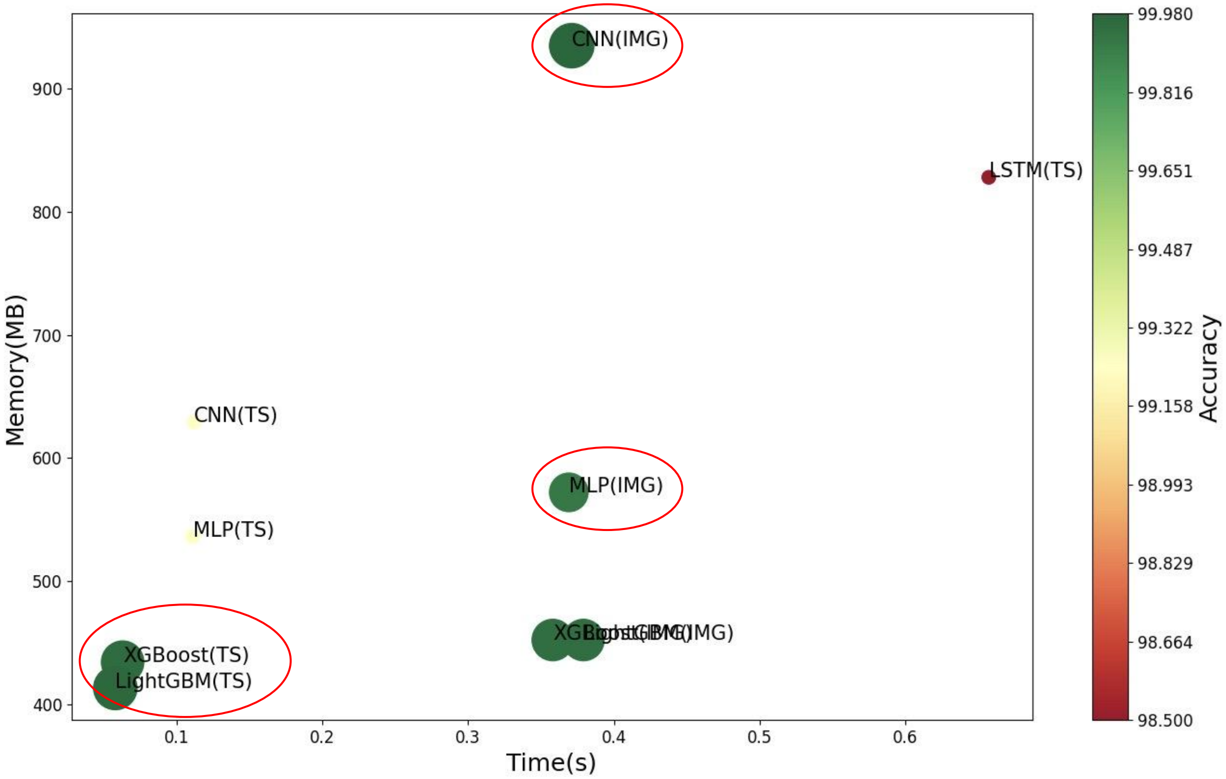
Model/Dataset	Time Series Data				Image Data			
Window size 1000	Accuracy	Precision	Recall	F1 Score	Accuracy	Precision	Recall	F1 Score
LSTM	98.50	99.65	97.30	98.46	N/A	N/A	N/A	N/A
XGBoost	99.96	99.99	99.93	99.96	99.95	99.99	99.91	99.95
LightGBM	99.97	100	99.95	99.97	99.95	99.99	99.91	99.95
MLP	99.24	99.29	99.20	99.23	99.92	99.95	99.90	99.92
CNN	99.24	99.02	99.47	99.25	99.98	99.99	99.96	99.98

Deployment Resource Requirement

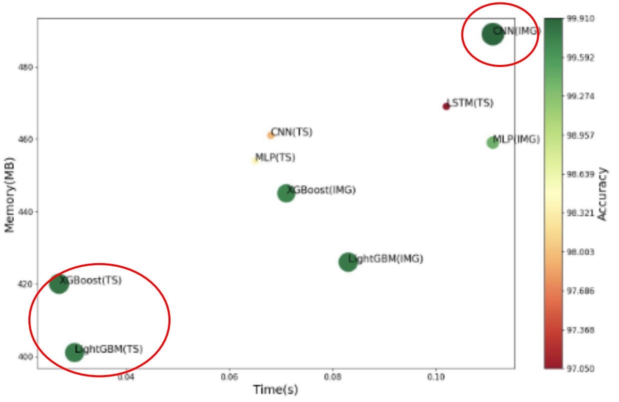
- Time requirement to process the data
- Prediction time requirement for classifier model
- Model memory usage during deployment

Classification Model	Data Processing (s)	Predict time (s)	Memory Usage (MB)
Window size 1000			
LGBM(TS)	0.051	0.007	413
XGBoost(TS)	0.048	0.015	434
LGBM(IMG)	0.341	0.038	452
MLP(IMG)	0.317	0.052	572
XGBoost(IMG)	0.305	0.053	452
CNN(IMG)	0.312	0.059	935
MLP(TS)	0.050	0.061	536
CNN(TS)	0.050	0.062	629
LSTM(TS)	0.050	0.607	828

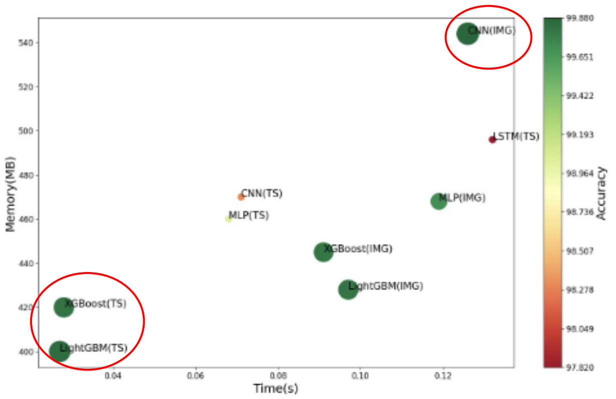
Model Performance vs Efficiency (Window size = 1000)



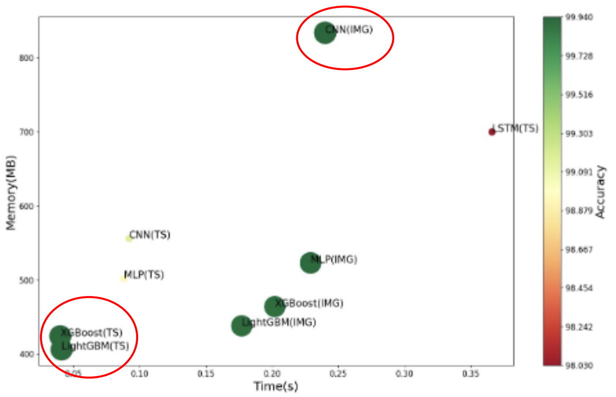
Model Performance vs Efficiency (50, 100, 500, 1000 window sizes)



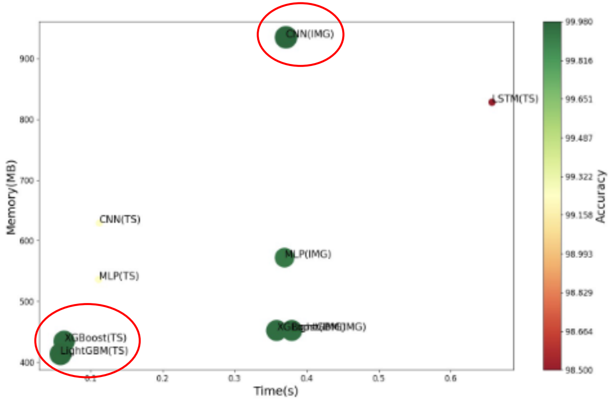
(a) Window Size of 50



(b) Window Size of 100



(c) Window Size of 500



(d) Window Size of 1000

Conclusion

- Explore state of the art models for ransomware detection using low-level hardware information
- Compare detection performance vs deployment cost
- CNN and gradient boosting model show exceptional detection capability
- LightGBM is the most efficient model interm of deployment cost for deployment

Q&A