1. Record Nr. UNINA9910584484003321

Titolo	Artificial intelligence for cybersecurity / / edited by Mark Stamp [and three others]
Pubbl/distr/stampa	Cham, Switzerland : , : Springer, , [2022] ©2022
ISBN	3-030-97087-6
Descrizione fisica	1 online resource (387 pages)
Collana	Advances in Information Security ; ; v.54
Disciplina	006.3
Soggetti	Machine learning
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di bibliografia	Includes bibliographical references.
Nota di contenuto	Intro -- Preface -- Contents -- About the Editors -- Part I Malware-Related Topics -- Generation of Adversarial Malware and Benign Examples Using Reinforcement Learning -- 1 Introduction -- 2 Background -- 2.1 Adversarial Machine Learning -- 2.1.1 Taxonomy -- 2.2 Reinforcement Learning -- 2.3 Portable Executable File Format -- 3 Implementation -- 3.1 Overview -- 3.2 Dataset -- 3.3 PE File Modifications -- 3.4 Target Classifier -- 3.5 Agent and Its Environment -- 4 Evaluation -- 4.1 Adversarial Malware Examples -- 4.2 Adversarial Benign Examples -- 5 Related Work -- 5.1 Gradient-Based Attacks -- 5.2 Reinforcement Learning-Based Attacks -- 5.3 Other Methods -- 6 Conclusion -- 6.1 Future Work -- References -- Auxiliary-Classifier GAN for Malware Analysis -- 1 Introduction -- 2 Related Work -- 3 Methodology -- 3.1 Data -- 3.2 AC-GAN -- 3.3 Evaluation Plan -- 3.3.1 CNN -- 3.3.2 ELM -- 3.4 Accuracy -- 4 Implementation -- 4.1 Dataset Analysis and Conversion -- 4.1.1 Datasets -- 4.2 AC-GAN Implementation -- 4.2.1 AC-GAN Generator -- 4.2.2 AC-GAN Discriminator -- 4.3 Evaluation Models -- 4.3.1 CNN Implementation -- 4.3.2 ELM Implementation -- 5 Experimental Results -- 5.1 AC-GAN Experiments -- 5.1.1 AC-GAN with 3232 Images -- 5.1.2 AC-GAN with 6464 Images -- 5.1.3 AC-GAN with 128128 Images -- 5.1.4 Summary of AC-GAN Results -- 5.2 CNN and ELM Experiments -- 5.2.1 CNN and ELM for 3232 Images -- 5.2.2 CNN and ELM for 6464 Images -- 5.2.3 CNN and ELM for 128128 Images -- 5.2.4 Discussion of CNN and ELM

Experiments -- 6 Conclusion and Future Work -- References -- Assessing the Robustness of an Image-Based Malware Classifier with Smali Level Perturbations Techniques -- 1 Introduction -- 2 Background and Related Works -- 2.1 Static Malware Analysis -- 2.2 Convolutional Neural Network -- 2.2.1 Convolution -- 2.2.2 Subsampling -- 2.2.3 Classification.
2.3 Dalvik VM and Dalvik EXecutable -- 2.4 Image-Based Malware Classification -- 3 Methodology -- 3.1 Untargeted Misclassification -- 4 Implementation and Experiments -- 5 Conclusion and Future Work -- References -- Detecting Botnets Through Deep Learning and Network Flow Analysis -- 1 Introduction -- 2 Background -- 2.1 Introduction to Botnets -- 2.2 Autocorrelation Analysis -- 2.3 Deep Neural Networks -- 3 Related Work -- 4 Dataset -- 4.1 CTU-13 Dataset Features -- 5 Proposed Methodology -- 5.1 Data Preprocessing Phase -- 5.1.1 Filtering Network Flow -- 5.1.2 Constructing Network Graph -- 5.1.3 Statistical Analysis of Edges -- 5.1.4 Autocorrelation Analysis -- 5.2 Deep Learning Phase -- 5.2.1 Stratified Lg-Fold Cross Validation -- 5.2.2 Define, Compile, and Fit the Neural Network -- 5.2.3 Model Evaluation -- 6 Results -- 7 Conclusions -- References -- Interpretability of Machine Learning-Based Results of Malware Detection Using a Set of Rules -- 1 Introduction -- 2 Related Works -- 3 Rule-Based Classification -- 3.1 From Trees to Rules -- 3.2 Rule-Learning Algorithms -- 4 Implementation of Rule-Based Classifiers -- 4.1 Decision List -- 4.2 I-REP -- 4.3 RIPPER -- 5 Experiments -- 5.1 Dataset Description -- 5.2 Data Splitting -- 5.3 Feature Transformation and Selection -- 5.4 Evaluation Metrics -- 5.5 Interpretability of Machine Learning Models -- 5.6 Measuring Performance of RBCs on ML Predictions -- 5.7 Interpreting ML Results Using RBCs -- 5.8 Pruning and Metrics -- 5.9 Does Order of the Rules Matter? -- 6 Conclusion and Future Work -- References -- Mobile Malware Detection Using Consortium Blockchain -- 1 Introduction -- 2 Use Case -- 3 Android Application Components -- 3.1 Activities -- 3.2 Services -- 3.3 Broadcast Receivers -- 3.4 Content Providers -- 4 Role in Malware Detection -- 5 The Blockchain Network -- 6 Related Works -- 7 Methodology.
7.1 APK Files -- 7.2 Trusted Server -- 7.3 Adding a Record -- 7.4 Members of the Consortium -- 7.5 Blockchain Ledger -- 7.6 Final Response -- 7.7 Technology Behind Blockchain Network -- 8 Implementation Details -- 8.1 Scenario 1 -- 8.2 Scenario 2 -- 8.3 Initializing Block for Unknown apk -- 8.4 Updating Block with Vote and Features -- 8.5 Setting the State of the apk After Counting All the Votes -- 9 Feature Extraction and Model Training -- 10 Dataset and Experimentation -- 11 Results -- 12 Conclusion -- References -- BERT for Malware Classification -- 1 Introduction -- 2 Related Work -- 3 Background -- 3.1 NLP Models -- 3.1.1 Word Embeddings -- 3.1.2 Word2Vec -- 3.1.3 BERT -- 3.2 Classifiers -- 3.2.1 Logistic Regression -- 3.2.2 SVM -- 3.2.3 Random Forests -- 3.2.4 MLP -- 4 Experiments and Results -- 4.1 Dataset -- 4.2 Methodology -- 4.3 Classifier Parameters -- 4.4 Logistic Regression Results -- 4.5 SVM Results -- 4.6 Random Forest Results -- 4.7 MLP Results -- 4.8 Further Analysis -- 4.9 Summary -- 5 Conclusions and Future Work -- References -- Machine Learning for Malware Evolution Detection -- 1 Introduction -- 2 Background -- 2.1 Malware -- 2.2 Related Work -- 2.3 Dataset -- 2.4 Learning Techniques -- 2.4.1 Hidden Markov Models -- 2.4.2 Word2Vec -- 2.4.3 HMM2Vec -- 2.4.4 Logistic Regression -- 3 Experiments and Results -- 3.1 Logistic Regression Experiments -- 3.2 Hidden Markov Model Experiments -- 3.3 HMM2Vec Experiments -- 3.4 Word2Vec Experiments -- 3.5 Discussion -- 4 Conclusion and

Future Work -- Appendix -- References -- Part II Other Security Topics -- Gambling for Success: The Lottery Ticket Hypothesis in Deep Learning-Based Side-Channel Analysis -- 1 Introduction -- 2 Background -- 2.1 Notation -- 2.2 Supervised Machine Learning in Profiling SCA -- 2.3 Leakage Models and Datasets -- 3 Related Works. 4 The Lottery Ticket Hypothesis (LTH) -- 4.1 Pruning Strategy -- 4.2 Winning Tickets in Profiling SCA -- 5 Experimental Results -- 5.1 Baseline Neural Networks -- 5.2 ASCAD with a Fixed Key -- 5.3 ASCAD with Random Keys -- 5.4 CHES CTF 2018 -- 5.5 General Observations -- 6 Conclusions and Future Work -- References -- Evaluating Deep Learning Models and Adversarial Attacks on Accelerometer-Based Gesture Authentication -- 1 Introduction -- 2 Related Work -- 3 Background -- 3.1 Support Vector Machines -- 3.2 1D Convolutional Neural Networks -- 3.3 Adversarial Strategy -- 3.3.1 Deep Convolutional Generative Adversarial Networks -- 4 Dataset -- 4.1 Data Collection -- 4.2 Data Preprocesssing -- 4.2.1 Feature Engineering -- 4.2.2 Time Series Resampling -- 5 Implementation -- 5.1 DC-GAN Structure -- 5.2 Adversarial Attack -- 6 Experiments and Results -- 6.1 SVM Results -- 6.2 1D-CNN Results -- 6.3 Adversarial Results -- 7 Conclusion and Future Work -- References -- Clickbait Detection for YouTube Videos -- 1 Introduction -- 2 Background -- 2.1 Related Work -- 2.1.1 Clickbait Detection -- 2.1.2 Fake News Detection -- 2.1.3 Forgery Detection -- 2.1.4 Hoax Detection -- 2.2 Natural Language Processing -- 2.3 Learning Techniques -- 2.3.1 Logistic Regression -- 2.3.2 Random Forest -- 2.3.3 Multilayer Perceptron -- 3 Implementation -- 3.1 Hardware and Software -- 3.2 Approach -- 3.3 Features -- 3.4 Dataset -- 3.5 Experiments -- 3.5.1 Experiment I: Logistic Regression with Word2Vec -- 3.5.2 Experiment II: Random Forest with Word2Vec -- 3.5.3 Experiment III: MLP with Word2Vec -- 3.5.4 Experiment IV: MLP with DropOut, Batch Normalization, and Word2Vec -- 3.5.5 Experiment V: MLP with BERT -- 3.5.6 Experiment VI: MLP with DistilBERT -- 4 Results -- 5 Conclusion and Future Works -- Appendix: Model Architectures -- References. Survivability Using Artificial Intelligence Assisted Cyber RiskWarning -- 1 Introduction -- 2 Related Work -- 3 Security Infringement Detection -- 3.1 Static Analysis of Code -- 3.2 Methodology -- 3.3 Results -- 3.4 Evaluation -- 4 Digital Twin Cyber Resilience Decision Support -- 4.1 Landscape Model Development -- 5 Semi-Markov Cloud Availability Model -- 6 Future Work -- 7 Conclusions -- References -- Machine Learning and Deep Learning for Fixed-Text Keystroke Dynamics -- 1 Introduction -- 2 Background -- 2.1 Keystroke Dynamics -- 2.2 Learning Techniques -- 2.2.1 Random Forest -- 2.2.2 Support Vector Machine -- 2.2.3 K-Nearest Neighbors -- 2.2.4 T-SNE -- 2.2.5 XGBoost -- 2.2.6 LSTM and Bi-LSTM -- 2.2.7 Convolutional Neural Network -- 2.2.8 Multi-Layer Perceptron -- 3 Previous Work -- 4 Dataset -- 5 Experiments and Results -- 5.1 Data Exploration -- 5.2 Classification Results -- 5.2.1 K-Nearest Neighbor Experiments -- 5.2.2 Random Forest Experiments -- 5.2.3 Support Vector Machine Experiments -- 5.2.4 XBGoost Experiments -- 5.2.5 Multilayer Perceptron Experiments -- 5.2.6 Convolutional Neural Network Experiments -- 5.2.7 Recurrent Neural Network Experiments -- 5.2.8 LSTM Experiments -- 5.3 Summary and Discussion -- 6 Conclusion and Future Work -- References -- Machine Learning-Based Analysis of Free-Text Keystroke Dynamics -- 1 Introduction -- 2 Background -- 2.1 Keystroke Dynamics -- 2.2 Previous Work -- 2.2.1 Distance Based Research -- 2.2.2 Machine Learning Based Research -- 3 Implementation -- 3.1 Dataset -- 3.2 Techniques Considered -- 3.2.1 BERT -- 3.2.2 CNN-GRU Model -- 4 Free-Text Experiments -- 4.1

Text-Based Classification -- 4.2 Keystroke Dynamics Models -- 4.2.1 Features -- 4.2.2 Parameter Tuning -- 4.2.3 Fine Tuning -- 4.2.4 GRU with Word Embedding -- 4.2.5 CNN-Transformer -- 4.2.6 CNN-GRU-Cross-Entropy-Loss -- 4.2.7 Rotation Subset.
4.2.8 Robustness.