1. Record Nr. UNINA9910502619303321

Titolo	Computer security - ESORICS 2021 : 26th European Symposium on Research in Computer Security, Darmstadt, Germany, October 4-8, 2021, Proceedings, Part I / / edited by Elisa Bertino, Haya Shulman, and Michael Waidner
Pubbl/distr/stampa	Cham, Switzerland : , : Springer, , [2021] ©2021
ISBN	3-030-88418-X
Descrizione fisica	1 online resource (798 pages)
Collana	Lecture Notes in Computer Science ; ; v.12972
Disciplina	005.82
Soggetti	Computer security Data encryption (Computer science)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Intro -- Preface -- Organization -- Keynotes -- Algorithms and the Law -- The Politics and Technology of (Hardware) Trojans -- Increasing Trust in ML Through Governance -- The Science of Computer Science: An Offensive Research Perspective -- Contents - Part I -- Contents - Part II -- Network Security -- More Efficient Post-quantum KEMTLS with Pre-distributed Public Keys -- 1 Introduction -- 1.1 Pre-distributed Keys -- 2 Preliminaries -- 3 KEMTLS with Pre-distributed Long-Term Keys -- 3.1 Proactive Client Authentication -- 4 Security Analysis -- 5 Instantiation and Evaluation -- 5.1 Choice of Primitives -- 5.2 Implementation -- 5.3 Handshake Sizes -- 5.4 Handshake Times -- 6 Discussion -- A KEMTLS -- References -- How to (Legally) Keep Secrets from Mobile Operators -- 1 Introduction -- 1.1 Our Contributions -- 1.2 Related Work -- 2 Preliminaries -- 3 LIKE Protocols -- 4 Security Model -- 5 Our Protocol -- 6 Security -- 7 Proof-of-Concept Implementation -- 8 Conclusion -- A Model Complements -- B Proof Sketches -- References -- A Formal Security Analysis of Session Resumption Across Hostnames -- 1 Introduction -- 2 Preliminaries -- 2.1 Building Blocks -- 2.2 Multi-Stage Key Exchange -- 3 Breaking the Security of Session Resumption Across Hostnames in TLS 1.3 -- 3.1 Modeling TLS 1.3 Session Resumption as an MSKE

Protocol -- 3.2 The Attack -- 4 Secure SRAH Protocols -- 4.1 Constructing Secure SRAH Protocols -- References -- Attacks -- Caught in the Web: DoS Vulnerabilities in Parsers for Structured Data -- 1 Introduction -- 2 Motivation -- 3 Characteristics of the Vulnerability -- 3.1 Topologies -- 3.2 Traversals -- 3.3 Triggers -- 4 Modelling the Analysis -- 4.1 Preliminaries -- 4.2 Analysis Specification -- 5 Experimental Setup and Evaluation -- 5.1 Approach -- 5.2 Implementation -- 5.3 Libraries for Analysis -- 5.4 Triggers or Entry Points.
5.5 Evaluation -- 6 Results and Discussion -- 6.1 PDF Vulnerabilities -- 6.2 Scalable Vector Graphics (SVG) Vulnerability -- 6.3 YAML Vulnerability -- 6.4 Newly Discovered Security Vulnerabilities -- 6.5 Threats to Validity -- 7 Related Work -- 7.1 Detecting Algorithmic Complexity Vulnerabilities -- 7.2 Traversals/Performance Bugs -- 8 Conclusion -- References -- PoW-How: An Enduring Timing Side-Channel to Evade Online Malware Sandboxes -- 1 Introduction -- 2 Background -- 2.1 Malware and Malware Analysis -- 2.2 PoW for Malware Analysis Evasion -- 2.3 Side-Channel Measurement -- 3 Our Approach: PoW-How -- 3.1 Threat Model -- 3.2 System Design -- 3.3 Performance Profiling -- 3.4 Threshold Estimation -- 3.5 Malware Integration and Testing -- 4 Evaluation -- 4.1 Threshold Estimation and PoW Algorithm Choice -- 4.2 Case Study: Known Malware -- 4.3 Case Study: Fresh Malware Sample -- 5 Security Analysis -- 6 Countermeasures -- 7 Discussion -- 7.1 Ethical Considerations -- 7.2 Bare-Metal Environments -- 7.3 Economical Denial of Sustainability -- 8 Related Work -- 9 Conclusion -- References -- Characterizing GPU Overclocking Faults -- 1 Introduction -- 1.1 Background -- 1.2 Related Work -- 1.3 Contributions -- 2 Preliminaries -- 2.1 CUDA -- 2.2 General GPU Setup -- 2.3 Overclocking -- 2.4 Attack Model -- 3 GPU Faults and Temperature Dependency -- 3.1 Setup -- 3.2 Initial Tests -- 3.3 Basics of Faults -- 3.4 The Relationship Between Faults and Temperature -- 4 Faults Boundaries and Values -- 4.1 The Boundaries of Faults -- 4.2 Byte-Flips and Bit-Flips -- 5 Memory Remnants and Transaction Size -- 5.1 The Basic Memory Transaction Size -- 5.2 A Model of the Memory Remnants Hypothesis: -- 5.3 Experimental Results -- 5.4 A Unified Fault Model -- 6 Countermeasures and Conclusions -- 6.1 Countermeasures -- 6.2 Conclusions -- A Appendix -- A.1 CUDA basics.
A.2 Future Work -- References -- Fuzzing -- ARIstoteles - Dissecting Apple's Baseband Interface -- 1 Introduction -- 2 Background and Related Work -- 2.1 Baseband Security Architecture -- 2.2 Baseband Interface Analysis Options on iOS -- 2.3 IOS Shared Libraries -- 3 Apple Remote Invocation Protocol -- 4 Fully-Automated Protocol Dissection -- 5 Automated Reverse-Engineering -- 5.1 Group and TLV Definitions -- 5.2 Type Definitions -- 5.3 Integrating Existing Dissectors -- 5.4 iOS Version Change Tracking -- 6 Fuzzing -- 6.1 Initial Fuzzing Considerations -- 6.2 Building and Optimizing Fuzzers -- 6.3 Crash Evaluation -- 7 Conclusion -- References -- webFuzz: Grey-Box Fuzzing for Web Applications -- 1 Introduction -- 1.1 Contributions -- 2 webFuzz -- 2.1 Instrumentation -- 2.2 Fuzzing Analysis -- 3 Bug Injection -- 3.1 Analysis and Injection -- 3.2 Bug Template -- 4 Evaluation -- 4.1 Code Coverage -- 4.2 Throughput -- 4.3 Vulnerability Detection -- 5 Limitations -- 6 Future Work -- 7 Related Work -- 8 Conclusion -- References -- My Fuzzer Beats Them All! Developing a Framework for Fair Evaluation and Comparison of Fuzzers -- 1 Introduction -- 2 Background -- 3 Statistical Evaluations -- 4 Problem Description and Related Work -- 5 Our Methodology -- 5.1 Comparing Fuzzers -- 5.2 Test Set Selection -- 5.3 Seed Sets --

5.4 Statistical Evaluation -- 5.5 Fuzzing Evaluation Setup -- 5.6 Test Runs -- 6 Experiments -- 6.1 Fuzzers -- 6.2 Seed Set -- 6.3 Run-Time -- 6.4 Number of Trials -- 6.5 Number of Bugs/Targets -- 6.6 Further Insights -- 7 Discussion -- 8 Conclusion -- References -- Malware -- Rope: Covert Multi-process Malware Execution with Return-Oriented Programming -- 1 Introduction -- 2 Background -- 2.1 Defenses for Systems and Applications -- 2.2 Distributed Malware -- 3 Challenges for Covert Distributed Malware -- 4 Rope -- 4.1 Architecture.
4.2 Loader Component -- 4.3 Chunk Crafting and ROP-TxF Layout -- 4.4 Bootstrap Component -- 4.5 Discussion -- 5 Implementation -- 6 Evaluation -- 7 Countermeasures and Wrap-Up -- References -- Towards Automating Code-Reuse Attacks Using Synthesized Gadget Chains -- 1 Introduction -- 2 Shortcomings of State-of-the-Art Approaches -- 3 Design -- 3.1 Gadgets -- 3.2 Logical Encoding -- 3.3 Preconditions and Postconditions -- 3.4 Formula Generation -- 3.5 Algorithm Configuration -- 4 Implementation -- 5 Evaluation -- 5.1 Setup -- 5.2 Finding a Chain -- 5.3 Real-World Applicability -- 5.4 Target-Specific Constraints -- 5.5 Chain Statistics -- 5.6 SGC's Configuration -- 6 Discussion -- 7 Related Work -- 8 Conclusion -- A Modeling -- B dnsmasq CVE-2017-14493 -- References -- Peeler: Profiling Kernel-Level Events to Detect Ransomware -- 1 Introduction -- 2 Related Work -- 3 Key Characteristics to Detect Ransomware -- 3.1 Application Contextual Behavior -- 3.2 Application Behavioral Characteristics -- 4 System Design -- 4.1 Overview -- 4.2 System Events Monitor -- 4.3 Malicious Commands Detector -- 4.4 Machine Learning-Based Classifier -- 5 Dataset Collection -- 5.1 Ransomware -- 5.2 Benign Applications -- 6 Evaluation -- 6.1 Detection Accuracy -- 6.2 Effectiveness in Detecting Abused Tools/utilities -- 6.3 Robustness Against Unseen Families -- 7 Conclusion -- A Dataset -- A.1 Ransomware families -- A.2 Benign applications -- References -- User Behaviour and Underground Economy -- Mingling of Clear and Muddy Water: Understanding and Detecting Semantic Confusion in Blackhat SEO -- 1 Introduction -- 2 Background -- 2.1 Search Engine Optimization (SEO) -- 2.2 Blackhat SEO -- 2.3 Semantic-Based Techniques -- 3 Semantic Confusion Detection -- 3.1 System Overview -- 3.2 Datasets -- 3.3 Data Processor -- 3.4 Semantic Analyzer -- 3.5 SEO Collector.
4 Implementation and Evaluation -- 4.1 Implementation -- 4.2 Evaluation -- 5 Measurement -- 5.1 Overview -- 5.2 SEO Domains -- 5.3 SEO Campaigns -- 5.4 Real-World Deployment -- 6 Practical Issues -- 7 Discussion -- 8 Related Work -- 9 Conclusion -- References -- An Explainable Online Password Strength Estimator -- 1 Introduction -- 1.1 Background -- 1.2 Contributions -- 2 Rank Estimation and Key Enumeration in Cryptographic Side-Channel Attacks -- 3 Multi-dimensional Models for Passwords -- 3.1 Overview -- 3.2 The Data Corpus -- 3.3 Selecting Dimensions -- 3.4 The Learning Phase -- 3.5 The Estimation Phase -- 3.6 Estimating the Ranks of Unleaked Password Parts -- 3.7 Performance -- 4 Usability of PESrank -- 4.1 A Proof of Concept Study -- 4.2 Explainability -- 5 Comparison with Existing Methods -- 5.1 Comparison to Cracker-Based and Neural Methods -- 5.2 Storage Requirements -- 6 Related Work -- 7 Conclusions -- A Additional Related Work -- A.1 Heuristic pure-estimator approaches -- A.2 Tweakable extensions and variations -- References -- Detecting Video-Game Injectors Exchanged in Game Cheating Communities -- 1 Introduction -- 2 Methodology -- 2.1 Data Collection -- 2.2 Post Analysis -- 2.3 Attachment Analysis -- 2.4 Injector Classifier -- 3 Results -- 3.1 Dataset Characterization -- 3.2 Injectors Classifier -- 3.3 Forum Analysis -- 4 Related Work -- 5