The Art Of Software Security Assessment Identifying And Avoiding Vulnerabilities Mark Dowd

Exploring Security in Software Architecture and Design
Designing Secure Software
How to Break Software Security
Software Security Engineering
Produktiv programmieren
Practical Core Software Security
PHP & MySQL von Kopf bis Fuss
Software Security -- Theories and Systems
Du darfst nicht alles glauben, was du denkst
Hacking & Security
Inside Anonymous
Kuckucksei
Software Security
Software Security Engineering
Clean Coder
Proceedings of Defining the State of the Art in Software Security Tools Workshop
The Art of Software Security Testing
The CERT C Secure Coding Standard
19 Deadly Sins of Software Security
Handbook of Software Reliability and Security Testing
Die Kunst der Anonymität im Internet
Die Geheimnisse des FBI
Methodisches Testen von Programmen
Engineering Safe and Secure Software Systems
Core Software Security
Angewandte Kryptographie
Hacking
The Art of Software Security Assessment
Mehr Hacking mit Python
CERT C Secure Coding Standard
Secrets & lies
Clean Code - Refactoring, Patterns, Testen und Techniken für sauberen Code
Kubernetes in Action
Die Kunst des Vertrauens
Basiswissen Sichere Software

Building in Security at Agile Speed
Building Secure Software
24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them
Die Kunst der Täuschung

FRIDA COOPER

Exploring Security in Software Architecture and Design Pearson Education
Sichere Software zeichnet sich dadurch aus, dass sie jedem möglichen Angriff standhalten können muss. Jeder Beteiligte im Softwareentwicklungsprozess sollte bewusst auf die Schaffung dieser Eigenschaft einer Software hinarbeiten. Dieses Buch vermittelt, welche Aspekte dabei zu berücksichtigen sind und zeigt für alle wichtigen Bereiche der Softwareentwicklung auf, was jeweils für Sicherheit getan werden kann - und muss. Es deckt den Lehrplan zum Certified Professional for Secure Software Engineering nach ISSECO-Standard ab, eignet sich zum Selbststudium und als Begleitliteratur zu Schulungen.
Designing Secure Software Addison-Wesley
Ob Sie wollen oder nicht – jede Ihrer Online-Aktivitäten wird beobachtet und analysiert Sie haben keine Privatsphäre. Im Internet ist jeder Ihrer Klicks für Unternehmen, Regierungen und kriminelle Hacker uneingeschränkt sichtbar. Ihr Computer, Ihr Smartphone, Ihr Auto, Ihre Alarmanlage, ja sogar Ihr Kühlschrank bieten potenzielle Angriffspunkte für den Zugriff auf Ihre Daten. Niemand kennt sich besser aus mit dem Missbrauch persönlicher Daten als Kevin Mitnick. Als von der US-Regierung ehemals meistgesuchter Computer-Hacker kennt er alle Schwachstellen und Sicherheitslücken des digitalen Zeitalters. Seine Fallbeispiele sind spannend und erschreckend: Sie werden Ihre Aktivitäten im Internet neu überdenken. Mitnick weiß aber auch, wie Sie Ihre Daten bestmöglich schützen. Er zeigt Ihnen anhand zahlreicher praktischer Tipps und Schritt-für-Schritt-Anleitungen, was Sie tun können, um online und offline anonym zu sein. Bestimmen Sie selbst über Ihre Daten. Lernen Sie, Ihre Privatsphäre im Internet zu schützen. Kevin Mitnick zeigt Ihnen, wie es geht. Hinterlassen Sie keine Spuren ● Sichere Passwörter festlegen und verwalten ● Mit dem Tor-Browser im Internet surfen, ohne Spuren zu hinterlassen ● E-Mails und Dateien verschlüsseln und vor fremden Zugriffen schützen ● Öffentliches WLAN, WhatsApp, Facebook & Co. sicher nutzen ● Sicherheitsrisiken vermeiden bei GPS, Smart-TV, Internet of Things und Heimautomation ● Eine zweite Identität anlegen und unsichtbar werden
How to Break Software Security CRC Press
In dieser brillanten Abhandlung, die mit philosophischen, vor allem spieltheoretischen Überlegungen ebenso zu überzeugen weiß wie mit fundierten wissenschaftlichen Erkenntnissen aus der Soziologie, Biologie und Anthroplogie, geht der IT-Sicherheitsexperte Bruce Schneier der Frage nach: Wieviel Vertrauen (der Individuen untereinander) braucht eine lebendige, fortschrittsorientierte Gesellschaft und wieviel Vertrauensbruch darf bzw. muss sie sich leisten?
Software Security Engineering Addison-Wesley Professional
The Art of Software Security AssessmentPearson Education
Produktiv programmieren dpunkt.verlag
Today's high-speed and rapidly changing development environments demand equally high-speed security practices. Still, achieving security remains a human endeavor, a core part of designing, generating and verifying software. Dr. James Ransome and Brook S.E. Schoenfield have built upon their previous works to explain that security starts with people; ultimately, humans generate software security. People collectively act through a particular and distinct set of methodologies, processes, and technologies that the authors have brought together into a newly designed, holistic, generic software development lifecycle facilitating software security at Agile, DevOps speed. —Eric. S. Yuan, Founder and CEO, Zoom Video Communications, Inc. It is essential that we embrace a mantra that ensures security is baked in throughout any development process. Ransome and Schoenfield leverage their abundance of experience and knowledge to clearly define why and how we need to build this new model around an understanding that the human element is the ultimate key to success. —Jennifer Sunshine Steffens, CEO of IOActive Both practical and strategic, Building in Security at Agile Speed is an invaluable resource for change leaders committed to building secure software solutions in a world characterized by increasing threats and uncertainty. Ransome and Schoenfield brilliantly demonstrate why creating robust software is a result of not only technical, but deeply human elements of agile ways of working. —Jorgen Hesselberg, author of Unlocking Agility and Cofounder of Comparative Agility The proliferation of open source components and distributed software services makes the principles detailed in Building in Security at Agile Speed more relevant than ever. Incorporating the principles and detailed guidance in this book into your SDLC is a must for all software developers and IT organizations. —George K Tsantes, CEO of Cyberphos, former partner at Accenture and Principal at EY Detailing the people, processes, and technical aspects of software security, Building in Security at Agile Speed emphasizes that the people element remains critical because software is developed, managed, and exploited by humans. This book presents a step-by-step process for software security that is relevant to today’s technical, operational, business, and development environments with a focus on what humans can do to control and manage the process in the form of best practices and metrics.
Practical Core Software Security MITP-Verlags GmbH & Co. KG
"... an engaging book that will empower readers in both large and small software development and engineering organizations to build security into their products. ... Readers are armed with firm solutions for the fight against cyber threats." —Dr. Dena Haritos Tsamitis. Carnegie Mellon University "... a must read for security specialists, software developers and software engineers. ... should be part of every security professional’s library." —Dr. Larry Ponemon, Ponemon Institute "... the definitive how-to guide for software security professionals. Dr. Ransome, Anmol Misra, and Brook Schoenfield deftly outline the procedures and policies needed to integrate real security into the

software development process. ...A must-have for anyone on the front lines of the Cyber War ..." —Cedric Leighton, Colonel, USAF (Ret.), Cedric Leighton Associates "Dr. Ransome, Anmol Misra, and Brook Schoenfield give you a magic formula in this book - the methodology and process to build security into the entire software development life cycle so that the software is secured at the source! " —Eric S. Yuan, Zoom Video Communications There is much publicity regarding network security, but the real cyber Achilles' heel is insecure software. Millions of software vulnerabilities create a cyber house of cards, in which we conduct our digital lives. In response, security people build ever more elaborate cyber fortresses to protect this vulnerable software. Despite their efforts, cyber fortifications consistently fail to protect our digital treasures. Why? The security industry has failed to engage fully with the creative, innovative people who write software. Core Software Security expounds developer-centric software security, a holistic process to engage creativity for security. As long as software is developed by humans, it requires the human element to fix it. Developer-centric security is not only feasible but also cost effective and operationally relevant. The methodology builds security into software development, which lies at the heart of our cyber infrastructure. Whatever development method is employed, software must be secured at the source. Book Highlights: Supplies a practitioner's view of the SDL Considers Agile as a security enabler Covers the privacy elements in an SDL Outlines a holistic business-savvy SDL framework that includes people, process, and technology Highlights the key success factors, deliverables, and metrics for each phase of the SDL Examines cost efficiencies, optimized performance, and organizational structure of a developer-centric software security program and PSIRT Includes a chapter by noted security architect Brook Schoenfield who shares his insights and experiences in applying the book's SDL framework View the authors' website at http://www.androidinsecurity.com/

PHP & MySQL von Kopf bis Fuss MITP-Verlags GmbH & Co. KG

Verhaltensregeln für professionelle Programmierer Erfolgreiche Programmierer haben eines gemeinsam: Die Praxis der Software-Entwicklung ist ihnen eine Herzensangelegenheit. Auch wenn sie unter einem nicht nachlassenden Druck arbeiten, setzen sie sich engagiert ein. Software-Entwicklung ist für sie eine Handwerkskunst. In Clean Coder stellt der legendäre Software-Experte Robert C. Martin die Disziplinen, Techniken, Tools und Methoden vor, die Programmierer zu Profis machen. Dieses Buch steckt voller praktischer Ratschläge und behandelt alle wichtigen Themen vom professionellen Verhalten und Zeitmanagement über die Aufwandsschätzung bis zum Refactoring und Testen. Hier geht es um mehr als nur um Technik: Es geht um die innere Haltung. Martin zeigt, wie Sie sich als Software-Entwickler professionell verhalten, gut und sauber arbeiten und verlässlich kommunizieren und planen. Er beschreibt, wie Sie sich schwierigen Entscheidungen stellen und zeigt, dass das eigene Wissen zu verantwortungsvollem Handeln verpflichtet. In diesem Buch lernen Sie: Was es bedeutet, sich als echter Profi zu verhalten Wie Sie mit Konflikten, knappen Zeitplänen und unvernünftigen Managern umgehen Wie Sie beim Programmieren im Fluss bleiben und Schreibblockaden überwinden Wie Sie mit unerbittlichem Druck umgehen und Burnout vermeiden Wie Sie Ihr Zeitmanagement optimieren Wie Sie für Umgebungen sorgen, in denen Programmierer und Teams wachsen und sich wohlfühlen Wann Sie Nein sagen sollten – und wie Sie das anstellen Wann Sie Ja sagen sollten – und was ein Ja wirklich bedeutet Großartige Software ist etwas Bewundernswertes: Sie ist leistungsfähig, elegant, funktional und erfreut bei der Arbeit sowohl den Entwickler als auch den Anwender. Hervorragende Software wird nicht von Maschinen geschrieben, sondern von Profis, die sich dieser Handwerkskunst unerschütterlich verschrieben haben. Clean Coder hilft Ihnen, zu diesem Kreis zu gehören. Über den Autor: Robert C. Uncle Bob Martin ist seit 1970 Programmierer und bei Konferenzen in aller Welt ein begehrter Redner. Zu seinen Büchern gehören Clean Code – Refactoring, Patterns, Testen und Techniken für sauberen Code und Agile Software Development: Principles, Patterns, and Practices. Als überaus produktiver Autor hat Uncle Bob Hunderte von Artikeln, Abhandlungen und Blogbeiträgen verfasst. Er war Chefredakteur bei The C++ Report und der erste Vorsitzende der Agile Alliance. Martin gründete und leitet die Firma Object Mentor, Inc., die sich darauf spezialisiert hat, Unternehmen bei der Vollendung ihrer Projekte behilflich zu sein.

Software Security -- Theories and Systems No Starch Press

Wer seine Brötchen mit Software-Entwicklung verdient, braucht Strategien, um besser, schneller und kostengünstiger zu programmieren. Dieses Buch bietet Ihnen erprobte Hilfsmittel, die Zeit sparen, Ihre Produktivität erhöhen, und die Sie unabhängig von der.

Du darfst nicht alles glauben, was du denkst Pearson Education

Learn how to destroy security bugs in your software from a tester's point-of-view. It focuses your security test on the common vulnerabilities--ther user interface, software dependencies, design, process and memory. (Midwest)

Hacking & Security De Gruyter Oldenbourg

As long as humans write software, the key to successful software security is making the software development program process more efficient and effective. Although the approach of this textbook includes people, process, and technology approaches to software security, Practical Core Software Security: A Reference Framework stresses the people element of software security, which is still the most important part to manage as software is developed, controlled, and exploited by humans. The text outlines a step-by-step process for software security that is relevant to today's technical, operational, business, and development environments. It focuses on what humans can do to control and manage a secure software development process using best practices and metrics. Although security issues will always exist, students learn how to maximize an organization's ability to minimize vulnerabilities in software products before they are released or deployed by building security into the development process. The authors have worked with Fortune 500 companies and have often seen examples of the breakdown of security development lifecycle (SDL) practices. The text takes an experience-based approach to apply components of the best available SDL models in dealing with the problems described above. Software security best practices, an SDL model, and framework are presented in this book. Starting with an overview of the SDL, the text outlines a model for mapping SDL best practices to the software development life cycle (SDLC). It explains how to use this model to build and manage a mature SDL program. Exercises and an in-depth case study aid students in mastering the SDL model. Professionals skilled in secure software development and related tasks are in tremendous demand today. The industry continues to experience exponential demand that should continue to grow for the foreseeable future. This book can benefit professionals as much as students. As they integrate the book's ideas into their software security practices, their value increases to their organizations, management teams, community, and industry. About the

Authors Dr. James Ransome, PhD, CISSP, CISM is a veteran of numerous chief information security officer (CISO), chief security officer (CSO), and chief production security officer (CPSO) roles, as well as an author and co-author of numerous cybersecurity books. Anmol Misra is an accomplished leader, researcher, author, and security expert with over 16 years of experience in technology and cybersecurity. Mark S. Merkow, CISSP, CISM, CSSLP has over 25 years of experience in corporate information security and 17 years in the AppSec space helping to establish and lead application security initiatives to success and sustainment.

Inside Anonymous mitp Verlags GmbH & Co. KG

Der Klassiker zum Thema Softeware-Test, bereits in der 7. Auflage! Dieses Buch hilft Ihnen, Kosten zu senken: durch eine praxisbezogene Anleitung zum Testen von Programmen. Es ist ein Handbuch zur Optimierung des methodischen Testens in der Praxis. Darüber hinaus werden auch ökonomische und psychologische Aspekte von Programmtests betrachtet, ebenso Marketinginformationen, Testwerkzeuge, High-Order-Testing, Fehlerbehebung und Codeinspektionen. Der Preis dieses Buches macht sich vielfach bezahlt, wenn es Ihnen geholfen hat, auch nur einen Fehler zu entdecken.

Kuckucksei Nova Science Pub Incorporated

What every software professional should know about security. Designing Secure Software consolidates Loren Kohnfelder’s more than twenty years of experience into a concise, elegant guide to improving the security of technology products. Written for a wide range of software professionals, it emphasizes building security into software design early and involving the entire team in the process. The book begins with a discussion of core concepts like trust, threats, mitigation, secure design patterns, and cryptography. The second part, perhaps this book’s most unique and important contribution to the field, covers the process of designing and reviewing a software design with security considerations in mind. The final section details the most common coding flaws that create vulnerabilities, making copious use of code snippets written in C and Python to illustrate implementation vulnerabilities. You’ll learn how to: • Identify important assets, the attack surface, and the trust boundaries in a system • Evaluate the effectiveness of various threat mitigation candidates • Work with well-known secure coding patterns and libraries • Understand and prevent vulnerabilities like XSS and CSRF, memory flaws, and more • Use security testing to proactively identify vulnerabilities introduced into code • Review a software design for security flaws effectively and without judgment Kohnfelder’s career, spanning decades at Microsoft and Google, introduced numerous software security initiatives, including the co-creation of the STRIDE threat modeling framework used widely today. This book is a modern, pragmatic consolidation of his best practices, insights, and ideas about the future of software.

Software Security S. Fischer Verlag

Most organizations have a firewall, antivirus software, and intrusion detection systems, all of which are intended to keep attackers out. So why is computer security a bigger problem today than ever before? The answer is simple--bad software lies at the heart of all computer security problems. Traditional solutions simply treat the symptoms, not the problem, and usually do so in a reactive way. This book teaches you how to take a proactive approach to computer security. Building Secure Software cuts to the heart of computer security to help you get security right the first time. If you are serious about computer security, you need to read this book, which includes essential lessons for both security professionals who have come to realize that software is the problem, and software developers who intend to make their code behave. Written for anyone involved in software development and use—from managers to coders—this book is your first step toward building more secure software. Building Secure Software provides expert perspectives and techniques to help you ensure the security of essential software. If you consider threats and vulnerabilities early in the devel-opment cycle you can build security into your system. With this book you will learn how to determine an acceptable level of risk, develop security tests, and plug security holes before software is even shipped. Inside you'll find the ten guiding principles for software security, as well as detailed coverage of: Software risk management for security Selecting technologies to make your code more secure Security implications of open source and proprietary software How to audit software The dreaded buffer overflow Access control and password authentication Random number generation Applying cryptography Trust management and input Client-side security Dealing with firewalls Only by building secure software can you defend yourself against security breaches and gain the confidence that comes with knowing you won't have to play the "penetrate and patch" game anymore. Get it right the first time. Let these expert authors show you how to properly design your system; save time, money, and credibility; and preserve your customers' trust.

Software Security Engineering MITP-Verlags GmbH & Co. KG

Developing secure software requires the integration of numerous methods and tools into the development process, and software design is based on shared expert knowledge, claims, and opinions. Empirical methods, including data analytics, allow extracting knowledge and insights from the data that organizations collect from their processes and tools, and from the opinions of the experts who practice these processes and methods. This book introduces the reader to the fundamentals of empirical research methods, and demonstrates how these methods can be used to hone a secure software development lifecycle based on empirical data and published best practices.

Clean Coder Createspace Independent Publishing Platform

State-of-the-Art Software Security Testing: Expert, Up to Date, and Comprehensive The Art of Software Security Testing delivers in-depth, up-to-date, battle-tested techniques for anticipating and identifying software security problems before the “bad guys” do. Drawing on decades of experience in application and penetration testing, this book’s authors can help you transform your approach from mere “verification” to proactive “attack.” The authors begin by systematically reviewing the design and coding vulnerabilities that can arise in software, and offering realistic guidance in avoiding them. Next, they show you ways to customize software debugging tools to test the unique aspects of any program and then analyze the results to identify exploitable vulnerabilities. Coverage includes Tips on how to think the way software attackers think to strengthen your defense strategy Cost-effectively integrating security testing into your development lifecycle Using threat modeling to prioritize testing based on your top areas of risk Building testing labs for performing white-, grey-, and black-box software testing Choosing and using the right tools for each testing project Executing today’s leading attacks, from fault injection to buffer overflows Determining which flaws are most likely to be exploited by real-world attackers

Proceedings of Defining the State of the Art in Software Security Tools Workshop Artech House

Wenn es um die Entwicklung leistungsfähiger und effizienter Hacking-Tools geht, ist Python für die

meisten Sicherheitsanalytiker die Sprache der Wahl. Doch wie genau funktioniert das? In dem neuesten Buch von Justin Seitz - dem Autor des Bestsellers "Hacking mit Python" - entdecken Sie Pythons dunkle Seite. Sie entwickeln Netzwerk-Sniffer, manipulieren Pakete, infizieren virtuelle Maschinen, schaffen unsichtbare Trojaner und vieles mehr. Sie lernen praktisch, wie man • einen "Command-and-Control"-Trojaner mittels GitHub schafft • Sandboxing erkennt und gängige Malware-Aufgaben wie Keylogging und Screenshotting automatisiert • Windows-Rechte mittels kreativer Prozesskontrolle ausweitet • offensive Speicherforensik-Tricks nutzt, um Passwort-Hashes abzugreifen und Shellcode in virtuelle Maschinen einzuspeisen • das beliebte Web-Hacking-Tool Burp erweitert • die Windows COM-Automatisierung nutzt, um einen Man-in-the-Middle-Angriff durchzuführen • möglichst unbemerkt Daten aus einem Netzwerk abgreift Eine Reihe von Insider-Techniken und kreativen Aufgaben zeigen Ihnen, wie Sie die Hacks erweitern und eigene Exploits entwickeln können.

The Art of Software Security Testing "O'Reilly Media, Inc."

›Kuckucksei‹ schildert bis ins Detail die hochdramatische Jagd nach deutschen Hackern, die in amerikanische Computernetze eingedrungen waren. Es ist der autobiografische Report eines amerikanischen Computercracks, der leidenschaftlich für die Sicherheit der Datennetze kämpft. (Dieser Text bezieht sich auf eine frühere Ausgabe.)

The CERT C Secure Coding Standard O'Reilly Germany

»Ich war dreißig Jahre depressiv. Ich muss damit leben. Und ich habe keinen Bock, das zu verheimlichen.« Kurt Krömer ist einer der beliebtesten und bekanntesten Komiker des Landes. In seiner Sendung »Chez Krömer« sprach er offen über seine schwere Depression und seine Zeit in der Tagesklinik und hat damit Millionen von Menschen erreicht. Alexander Bojcan ist 47 Jahre alt, trockener Alkoholiker, alleinerziehender Vater und er war jahrelang depressiv. Auf der Bühne und im Fernsehen spielt er Kurt Krömer. Er will sich nicht länger verstecken. »Du darfst nicht alles glauben, was Du denkst« ist der schonungslos offene und gleichzeitig lustige Lebensbericht eines Künstlers, von dem die Öffentlichkeit bisher nicht viel Privates wusste. Alexander Bojcan bricht ein Tabu und das tut er nicht um des Tabubrechens willen, sondern um Menschen zu helfen, die unter Depressionen leiden oder eine ähnliche jahrelange Ärzteodyssee hinter sich haben wie er selbst. Dieses Buch wirbt für einen offenen Umgang mit psychischen Krankheiten und ist gleichzeitig kein Leidensbericht, sondern eine komische und extrem liebenswerte Liebeserklärung an das Leben und die Kunst. Ein großes, ein großartiges Buch. »Und ab dafür«, würde Kurt Krömer sagen.

19 Deadly Sins of Software Security Kiepenheuer & Witsch

Willkommen in der New Economy, der Welt der digitalen Wirtschaft. Informationen sind leichter zugänglich als je zuvor. Die Vernetzung wird dicher, und digitale Kommunikation ist aus den Unternehmen nicht mehr wegzudenken. Die Begeisterung für die Technologie hat jedoch Ihren Preis: Die Zahl der Sicherheitsrisiken nimmt ständig zu. Die neuen Gefahren, die mit dem E-Business verknüpft sind, müssen den Unternehmen weltweit aber erst klar werden. Dieses Buch ist ein erster Schritt in diese Richtung. Bruce Schneier, anerkannter Experte im Bereich Kryptographie, erklärt, was Unternehmen über IT-Sicherheit wissen müssen, um zu überleben und wettbewerbsfähig zu bleiben. Er deckt das gesamte System auf, von den Ursachen der Sicherheitslücken bis hin zu den Motiven, die hinter böswilligen Attacken stehen. Schneier zeigt Sicherheitstechnologien und deren Möglichkeiten, aber auch deren Grenzen auf. Fundiert und anschaulich zugleich behandelt dieser praktische Leitfaden: - Die digitalen Bedrohungen und Angriffe, die es zu kennen gilt - Die derzeit verfügbaren Sicherheitsprodukte und -prozesse - Die Technologien, die in den nächsten Jahren interessant werden könnten - Die Grenzen der Technik - Das Vorgehen, um Sicherheitsmängel an einem Produkt offenzulegen - Die Möglichkeiten, existierende Risiken in einem Unternehmen festzustellen - Die Implementierung einer wirksamen Sicherheitspolitik Schneiers Darstellung der digitalen Welt und unserer vernetzten Gesellschaft ist pragmatisch, interessant und humorvoll. Und sie ermöglicht es dem Leser, die vernetzte Welt zu verstehen und sich gegen ihre Bedrohungen zu wappnen. Hier finden Sie die Unterstützung eines Experten, die Sie für die Entscheidungsfindung im Bereich IT-Sicherheit brauchen.

Handbook of Software Reliability and Security Testing McGraw-Hill Osborne Media

For more than the last three decades, the security of software systems has been an important area of computer science, yet it is a rather recent general recognition that technologies for software security are highly needed. This book assesses the state of the art in software and systems security by presenting a carefully arranged selection of revised invited and reviewed papers. It covers basic aspects and recently developed topics such as security of pervasive computing, peer-to-peer systems and autonomous distributed agents, secure software circulation, compilers for fail-safe C language, construction of secure mail systems, type systems and multiset rewriting systems for security protocols, and privacy issues as well.