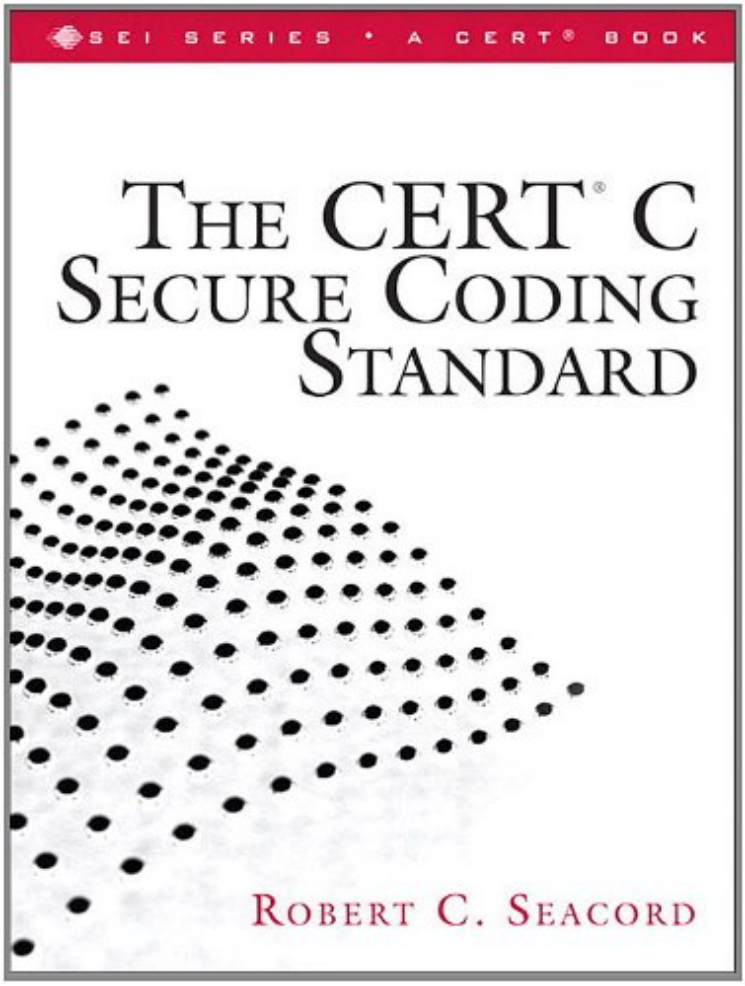
The CERT C Secure Coding Standard (SEI Series in Software Engineering)

Von Robert C. Seacord
*ePub | *DOC | audiobook | ebooks | Download PDF*



Produktinformation -Verkaufsrang: #884149 in eBooksVerffentlicht am: 2008-10-14Erscheinungsdatum: 2008-10-14File Name: B004Y4UTB8 | File size: 50.Mb

Von Robert C. Seacord : The CERT C Secure Coding Standard (SEI Series in Software Engineering) before purchasing it in order to gage whether or not it would be worth my time, and all praised The CERT C Secure Coding Standard (SEI Series in Software Engineering):

KundenrezensionenHilfreichste Kundenrezensionen0 von 0 Kunden fanden die folgende Rezension hilfreich. Trocken, aber sehr ntzlichVon Wolfgang WilhelmBevor Sie diese Rezension lesen, sollten Sie wissen, dass ich mit C nur als "zweite" Sprache arbeite; ich bevorzuge Perl. Es gibt aber auch Gelegenheiten, bei denen ich C verwende. Ich bin daher nicht der Experte in dieser Sprache.Zielgruppe des Buchs sind Programmierer, die hauptschlich oder ausschlielich in C programmieren. Es werden nicht nur die bekannten Fehler aus dem Bericht Speicher und Zeiger angesprochen, sondern auch die vielen anderen Kleinigkeiten, die sonst schief gehen knnen.Dazu gibt es eine

Beurteilung der Schwere und Wahrscheinlichkeit der Fehler.Die Einteilung des Buches erfolgt thematisch, z.B. Fehler beim Nutzen des Prprozessors oder bei Speicherroutinen.Was dem Autor wirklich in vielen Fllen gut gelingt, ist die trockene Materie "sicher programmieren" an den Leser zu bringen. Ich konnte noch einiges lernen, was ich so alles falsch gemacht habe.Das Buch bekommt nur vier Sterne, weil es eben in einigen Fllen trocken bleibt. Was mir wirklich abgeht, ist die themenbergreifende Sicht, z.B. eine Liste mit den nach Wahrscheinlichkeit sortierten Fehlern.0 von 2 Kunden fanden die folgende Rezension hilfreich. The CERT C Secure Coding StandardVon David SingletonDas Buch hatte meine Erwartungen getroffen. Ein sehr ntzliches Buch ber wie man richtig Code schreiben soll.

KurzbeschreibungIm an enthusiastic supporter of the CERT Secure Coding Initiative. Programmers have lots of sources of advice on correctness, clarity, maintainability, performance, and even safety. Advice on how specific language features affect security has been missing. The CERT C Secure Coding Standard fills this need. Randy Meyers, Chairman of ANSI C For years we have relied upon the CERT/CC to publish advisories documenting an endless stream of security problems. Now CERT has embodied the advice of leading technical experts to give programmers and managers the practical guidance needed to avoid those problems in new applications and to help secure legacy systems. Well done! Dr. Thomas Plum, founder of Plum Hall, Inc. Connectivity has sharply increased the need for secure, hacker-safe applications. By combining this CERT standard with other safety guidelines, customers gain all-round protection and approach the goal of zero-defect software. Chris Tapp, Field Applications Engineer, LDRA Ltd. Ive found this standard to be an indispensable collection of expert information on exactly how modern software systems fail in practice. It is the perfect place to start for establishing internal secure coding guidelines. You wont find this information elsewhere, and, when it comes to software security, what you dont know is often exactly what hurts you. John McDonald, coauthor of The Art of Software Security Assessment Software security has major implications for the operations and assets of organizations, as well as for the welfare of individuals. To create secure software, developers must know where the dangers lie. Secure programming in C can be more difficult than even many experienced programmers believe. This book is an essential desktop reference documenting the first official release of The CERT C Secure Coding Standard . The standard itemizes those coding errors that are the root causes of software vulnerabilities in C and prioritizes them by severity, likelihood of exploitation, and remediation costs. Each guideline provides examples of insecure code as well as secure, alternative implementations. If uniformly applied, these guidelines will eliminate the critical coding errors that lead to buffer overflows, format string vulnerabilities, integer overflow, and other common software vulnerabilities. KurzbeschreibungIm an enthusiastic supporter of the CERT Secure Coding Initiative. Programmers have lots of sources of advice on correctness, clarity, maintainability, performance, and even safety. Advice on how specific language features affect security has been missing. The CERT C Secure Coding Standard fills this need. Randy Meyers, Chairman of ANSI C For years we have relied upon the CERT/CC to publish advisories documenting an endless stream of security problems. Now CERT has embodied the advice of leading technical experts to give programmers and managers the practical guidance needed to avoid those problems in new applications and to help secure legacy systems. Well done! Dr. Thomas Plum, founder of Plum Hall, Inc. Connectivity has sharply increased the need for secure, hacker-safe applications. By combining this CERT standard with other safety guidelines, customers gain all-round protection and approach the goal of zero-defect software. Chris Tapp, Field Applications Engineer, LDRA Ltd. Ive found this standard to be an indispensable collection of expert information on exactly how modern software systems fail in practice. It is the perfect place to start for establishing internal secure coding guidelines. You wont find this information elsewhere, and, when it comes to software security, what you dont know is often exactly what hurts you. John McDonald, coauthor of The Art of Software Security Assessment Software security has major implications for the operations and assets of organizations, as well as for the welfare of individuals. To create secure software, developers must know where the dangers lie. Secure programming in C can be more difficult than even many experienced programmers believe. This book is an essential desktop reference documenting the first official release of The CERT C Secure Coding Standard . The standard itemizes those coding errors that are the root causes of software vulnerabilities in C and prioritizes them by severity, likelihood of exploitation, and remediation costs. Each guideline provides examples of insecure code as well as secure, alternative implementations. If uniformly applied, these guidelines will eliminate the critical coding errors that lead to buffer overflows, format string vulnerabilities, integer overflow, and other common software vulnerabilities. Synopsis "I'm an enthusiastic supporter of the CERT Secure Coding Initiative. Programmers have lots of sources of advice on correctness, clarity, maintainability, performance, and even safety. Advice on how specific language features affect security has been missing. The CERT(R) C Secure Coding Standardfills this need." --Randy Meyers, Chairman of ANSI C "For years we have relied upon the CERT/CC to publish advisories documenting an endless stream of security problems. Now CERT has embodied the advice of leading technical experts to give programmers and managers the practical guidance needed to avoid those problems in new applications and to help

secure legacy systems. Well done!" --Dr. Thomas Plum, founder of Plum Hall, Inc. "Connectivity has sharply increased the need for secure, hacker-safe applications. By combining this CERT standard with other safety guidelines, customers gain all-round protection and approach the goal of zero-defect software." --Chris Tapp, Field Applications Engineer, LDRA Ltd. "I've found this standard to be an indispensable collection of expert information on exactly how modern software systems fail in practice.It is the perfect place to start for establishing internal secure coding guidelines.You won't find this information elsewhere, and, when it comes to software security, what you don't know is often exactly what hurts you." --John McDonald, coauthor of The Art of Software Security Assessment Software security has major implications for the operations and assets of organizations, as well as for the welfare of individuals. To create secure software, developers must know where the dangers lie. Secure programming in C can be more difficult than even many experienced programmers believe. This book is an essential desktop reference documenting the first official release of The CERT(R) C Secure Coding Standard. The standard itemizes those coding errors that are the root causes of software vulnerabilities in C and prioritizes them by severity, likelihood of exploitation, and remediation costs. Each guideline provides examples of insecure code as well as secure, alternative implementations.If uniformly applied, these guidelines will eliminate the critical coding errors that lead to buffer overflows, format string vulnerabilities, integer overflow, and other common software vulnerabilities.