
KeepVid Music Tag Editor 2.0.0.3

Vorwort

Mit der wachsenden Abhängigkeit unserer Gesellschaft von der Zuverlässigkeit informationstechnischer Systeme (IT) gewinnen Fragen der IT-Sicherheit an Bedeutung. Während bisher vorrangig präventive Maßnahmen und Mechanismen im Vordergrund standen, wird zunehmend deutlich, dass IT-Sicherheit nicht allein durch Prävention erreichbar ist. Vielmehr stellt Prävention einen Grundpfeiler dar, neben dem ergänzend die reaktiven Aspekte der IT-Sicherheit stehen.

Die Fachgruppe SIDAR (Security - Intrusion Detection and Response) des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. fokussiert die reaktiven Aspekte der IT-Sicherheit sowie deren Umfeld. Mit dem Workshop DIMVA 2004 veranstaltet die FG SIDAR erstmals im deutschsprachigen Raum einen Workshop, der die Themen Intrusion Detection, Malware-Bekämpfung und Verwundbarkeitsanalyse in den Mittelpunkt stellt.

Der Workshop richtet sich an Personen und Organisationen, die in diesen Themenbereichen in Industrie, Dienstleistung, Verwaltung und Wissenschaft tätig sind. Ziel des Workshops ist es, sowohl den kommunikativen Austausch zu fördern, als auch reaktive Aspekte der IT-Sicherheit stärker in das Blickfeld der Öffentlichkeit zu rücken.

Dieser Band enthält ausgewählte Beiträge, die von den Autoren auf dem Workshop am 6. und 7. Juli in Dortmund (Deutschland) präsentiert wurden. Das Programmkomitee erhielt 41 Beiträge von Autoren aus zwölf Ländern und drei Kontinenten. 78% der Beiträge stammen von Autoren aus Europa. Die meisten Beiträge wurden von Autoren aus den folgenden Ländern eingereicht: Deutschland (26), USA (4), Vereinigte Arabische Emirate (3), Iran (2) und Schweiz (2). Jeder Beitrag wurde sorgfältig von mindestens drei Mitgliedern des Programmkomitees oder zusätzlichen Experten begutachtet und nach den folgenden Kriterien bewertet: wissenschaftliche Neuheit, Bedeutung für das Gebiet und technische Qualität.

Das Programmkomitee hat insgesamt 19 Beiträge zur Präsentation auf dem Workshop ausgewählt (46%). Davon wurden 14 Beiträge zur Veröffentlichung in diesem Band akzeptiert (34%) und fünf Kurzbeiträge werden auf der Web-Seite des Workshops veröffentlicht (12%). Die akzeptierten Beiträge wurden von den Autoren vor der Veröffentlichung überarbeitet. Die überarbeiteten Beiträge wurden keiner weiteren Begutachtung unterzogen, und die Autoren tragen die Verantwortung für den Inhalt ihrer Beiträge.

Das Workshop-Programm umfasste neue theoretische und praktische Ansätze und Resultate aus der Forschung sowie Erfahrungsberichte zum Schwerpunktthema Intrusion Detection und zu den Themen Honeypots, Verwundbarkeiten und Malware-Bekämpfung. Zum Auftakt des Workshop-Programms hielt Hans-Michael Hepp (Intelligent Risk Solutions) den Keynote-Vortrag "Verfahren der Transaktionsanalyse am Beispiel der Missbrauchsfrüherkennung im Kreditkartengeschäft". Die Präsentationen und die Kurzbeiträge sind auf der Web-Seite des Workshops verfügbar:
<http://www.gi-fg-sidar.de/dimva2004/>

Wir danken allen, die zum Gelingen des Workshops beigetragen haben, insbesondere den Autoren, dem eingeladenen Redner, dem Programmkomitee und den Gutachtern.

Juli 2004

Ulrich Flegel, Michael Meier

DOWNLOAD: <https://byltly.com/2iq71y>



Q: ASP.NET Forms authentication security issues Possible Duplicate: Should I use ASP.NET Forms authentication? I have a webforms application with a code behind that uses Forms Authentication. Currently users must click a log-in button in order to use the application. The application has a config file that defines how the application works. The application has a login form and a database of users. Once the user logs in the database is updated and the user can start working with the application. Is there a problem with this? I'm concerned that a user could log into my application and find out all the information about the application that is stored in the config file. Is there a way to secure the information in the config file? A: Forms Authentication is intended to protect against someone gaining access to your application by means of a username/password, so it will protect against them reading the config file. As you say, the config file would be a starting point for someone who wants to learn about the application. So if they gain access to that then they can start to learn more about it. But from a security point of view, you should be very confident that the config file is not a starting point for security issues. What your application is doing is logging the user in, and in the form of what you have done, the users credentials are being stored in the config file. The only problem with this is that the credentials could be stored in plain text and anyone reading the config file could be able to log in as the user. To prevent this you should encrypt the data, which means that you need to get the users credentials somehow before you store them in the config file, and this means that you need to authenticate the user. If you look at Authentication.cs in your code behind, you will see how easy it is to do this: `if (Membership.ValidateUser(username, password)) { FormsAuthentication.SetAuthCookie(username, true); }` The code inside the if statement will only be executed if the user is valid. You need to make sure that you get this value from somewhere, and ideally you should store it inside the config file so that it can be used in all your pages and all your code behind files. If you don't do this, then at the very least you should not be storing the user credentials in the 82157476af

Related links:

[Crime Patrol 2 Drug Wars American Lasergames 2003 Version PC warhammerageofsigmarorderbattletomeseraphonpdf](#)
[Everything Explained For The Professional Pilot Pdf Download Torrent Free Torrent 220](#)