Anne Koziolek, Ina Schaefer, Christoph Seidl (Hrsg.): Software Engineering 2021,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2021 101

Automated Implementation of Windows-related Security-Configuration Guides

Patrick Stöckle[1] Bernd Grobauer[2] Alexander Pretschner[3]

Abstract: Dieser Vortrag wurde auf der 35. IEEE/ACM International Conference on Automated Software Engineering (ASE) präsentiert. Unsicher konfigurierte Geräte stellen ein großes Sicherheitsproblem dar. Eine Möglichkeit, dieses Problem zu lösen, sind öffentlich verfügbare und standartisierte Sicherheitskonfigurationsrichtlinien. Dieser Ansatz birgt jedoch die Schwierigkeit, dass Administratoren auf Basis der Anleitungen in diesen Richtlinien ihre Systeme manuell sichern müssen. Dieses manuelle Sichern ist teuer und fehleranfällig. In unserem Beitrag präsentieren wir einen Ansatz, mit dem wir Richtlinien für Windows-Systeme automatisiert anwenden können. Dafür wenden wir Techniken der Sprachverarbeitung an. Im ersten Teil unserer Evaluation können wir anhand einer öffentlichen Richtlinie für Windows 10 zeigen, dass unser Ansatz für 83% der Regeln keinerlei menschliche Interaktion benötigt. Im zweiten Teil zeigen wir anhand von 12 öffentlichen Richtlinien mit über 2000 Regeln, dass unser Ansatz die Regeln zu 97% korrekt anwendet. So wird die sichere Konfiguration von Windows-Systemen einfacher und wir hoffen, dass dies zukünftig zu weniger Sicherheitsvorfällen führen wird.

Keywords: Sicherheit; Konfigurationsmanagement; Computerlinguistik

Fehlkonfigurationen verringern die Sicherheit eines Systems, indem sie Schwachstellen einführen, die oft schwer aufzuspüren sind. Administratoren zufolge gibt es einen Hauptfaktor dafür: mangelndes Wissen. [Di18] Eine Möglichkeit, mit diesem Problem umzugehen, ist das Verwenden von bestehenden Sicherheitskonfigurationsrichtlinien. Diese bestehen aus Regeln für ein bestimmtes Softwaresystem wie Windows 10. Jede Regel erklärt, welche Einstellung auf welchen Wert gesetzt werden sollte, um das System sicherer zu machen, und warum wir sie anwenden sollte. Bekannte Herausgeber solcher Richtlinien sind das Center for Internet Security (CIS) oder die Defense Information Systems Agency (DISA).

Die Herausgeber veröffentlichen ihre Richtlinien in Formaten wie PDF und im *Extensible Configuration Checklist Description Format (XCCDF)*, das Teil des *Security Content Automation Protocol (SCAP)* ist. Obwohl XCCDF als maschinenlesbares Format konzipiert ist, sind die Anweisungen zur Implementierung der Sicherheitseinstellungen nur in menschenlesbarer Form enthalten. Die vorhandenen Richtlinien lösen zwar das Problem des mangelnden Wissens, bringen aber eine neue Herausforderung mit sich: Automatische Umsetzungen sind im SCAP-Standard nicht spezifiziert. Die Herausgeber umgehen diese Hürde manchmal, indem sie zusätzliche Artefakte wie Skripte oder Backup-Dateien zur

[1] Technische Universität München (TUM), Boltzmannstr. 3, 85748 Garching b. München patrick.stoeckle@tum.de
[2] Siemens AG bernd.grobaueratsiemens.com
[3] Technische Universität München (TUM), Boltzmannstr. 3, 85748 Garching b. München alexander.pretschner@tum.de

doi:10.18420/SE2021_38

Verfügung stellen. Dies ist auf drei Arten problematisch: Erstens gibt es solche Artefakte nicht für alle Richtlinien. Zweitens werden die Richtlinien häufig aktualisiert und die Artefakte müssen auf Herausgeber-Seite oft und manuell aktualisiert werden. Drittens wird bei eigenständigen Artefakten für die Implementierung das Anpassen (tailoring) von Richtlinien umständlich und fehleranfällig. Eine einfache und leichte Anpassung ist jedoch unerlässlich, da Richtlinien von CIS oder DISA nie ohne Anpassungen in der eigenen Organisation umgesetzt werden.

Unsere für Windows-Betriebssysteme und -Anwendungen realisierte Lösung für dieses Problem besteht aus drei Schritten. Zuerst verarbeiten wir die Dateien, die definieren, welche Einstellungen auf einem Windows-basierten System existieren, und speichern das Ergebnis, um während der Umsetzung darauf zugreifen zu können. Zweitens nutzen wir Techniken der Computerlinguistik, um die Einstellungen und die geforderten Werte aus den Richtlinien zu extrahieren. Wir verwenden die Informationen des ersten Schritts, um zu überprüfen, ob die extrahierte Einstellung existiert und ob der extrahierte Wert eine gültige Eingabe für diese Einstellung ist. So können wir das Risiko falsch extrahierter Werte auf ein Minimum reduzieren. Drittens übersetzen wir die Einstellungen und Werte unter Verwendung der Informationen aus dem ersten Schritt in ihre tatsächliche Umsetzung. Unsere Beiträge sind:

- eine Proof-of-Concept-Implementierung unseres Ansatzes.
- eine Schritt-für-Schritt-Dokumentation unseres Ansatzes unter Verwendung der DISA Windows Server 2016 Richtlinie[4] und eine aktualisierte Version für 2019[5]. Hierbei können wir zeigen, dass unser Ansatz 83% der Regeln ohne manuellen Aufwand umsetzen kann.
- eine Evaluation unseres Ansatzes unter Verwendung bestehender Richtlinien von DISA und CIS mit über 2000 Regeln[6]; unser Ansatz setzt die gegebenen Regeln zu 97% korrekt um.

Durch unseren Ansatz wird die sichere Konfiguration von Windows-Systemen deutlich einfacher. Wir hoffen, dass in Zukunft mehr Administratoren ihre Systeme sicher konfigurieren werden und so das Risiko von Sicherheitsvorfällen sinkt.

Literatur

[Di18] Dietrich, C.; Krombholz, K.; Borgolte, K.; Fiebig, T.: Investigating System Operators' Perspective on Security Misconfigurations. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. CCS '18, ACM, Toronto, Canada, S. 1272–1289, 2018, ISBN: 978-1-4503-5693-0, URL: http://doi.acm.org/10.1145/3243734.3243794.

[4] swh:1:dir:c3803619f51702199b19405547e2be2f2f55bdd2
[5] swh:1:dir:13ffd9d2566c64afdedd414336a95a35605392d7
[6] swh:1:dir:b5c15f48b2c288f58533c9354bea3703ffbbb0dd