A. Weisbecker, M. Burmester & A. Schmidt (Hrsg.): Mensch und Computer 2015 Workshopband, Stuttgart: Oldenbourg Wissenschaftsverlag, 2015, S. 657-665.

Usable-Policy-Management in kollaborativen Szenarien - Herausforderungen und Chancen

Sascha Wagner[1], Dominic Heutelbeck[2]

Lehrgebiet Multimedia und Internetanwendungen, Fakultät für Mathematik und Informatik, Fernuniversität Hagen[1]
FTK e.V. Forschungsinstitut für Telekommunikation und Kooperation[2]

Zusammenfassung

Der Bedarf an sicheren und gleichermaßen gebrauchstauglichen Systemen steigt. Durch wachsende Komplexität und Dynamik der Nutzungsumgebungen wird die Ausbalancierung zwischen Security- und Usability-Zielen anspruchsvoll und herausfordernd. Kollaborative Arbeitsumgebungen und Szenarien stellen zudem zusätzliche Anforderungen, wie Flexibilität und Agilität, an die Gestaltung der einzusetzenden Systeme. Ein in der wissenschaftlichen Literatur noch unzureichend betrachteter Bereich ist hierbei das Policy-Management[1] in kollaborativen Szenarien. Dieser Beitrag stellt in diesem Spannungsfeld allgemeine Gestaltungsempfehlungen an Usable-Security-Systemen vor, analysiert aktuelle Lösungen im Bereich des Policy-Managements und leitet daraus Strategien, Forschungsfragen und einen Lösungsansatz zur Abbildung eines gebrauchstauglichen Policy-Managements in kollaborativen Szenarien ab.

1 Einleitung

Eine in der Literatur oft unzureichend beachtete Ebene im Umfeld von Usable Security ist die Abbildung von organisatorischen Zugriffsrichtlinien (engl. Access-Control-Policies) zur Durchsetzung von Security-Zielen innerhalb von Unternehmen. Ferner steigt der Bedarf an flexiblen Kooperationen und Kollaborationen zum Austausch und zur Generierung von Wissen. Mit diesen Zunahmen verbunden, steigt auch der Anspruch an Möglichkeiten zur effektiven und leicht umzusetzenden Abbildung und Nutzung von Zugriffsregeln.

[1] Policy-Management steht in dieser Arbeit stellvertretend für Access-Control-Policy-Management.

2 Policy-Management

Policy-Management ist ein essentieller Bestandteil jeder vertrauensvollen Zusammenarbeit. Es umfasst nach Johnson primär die folgenden drei Schritte (Johnson 2012):

- Initiale Erstellung der Zugriffsregeln
- Interpretation der Auswirkungen
- Durchführung von Änderungen

Diese Punkte können je nach Betrachtungsweise um Implementierung und Verwaltung erweitert werden. Innerhalb der einzelnen Phasen sind oft verschiedene Akteure an den jeweiligen Teilschritten, wie Risikobewertung, Entscheidungsfindung oder Abnahme, beteiligt. Um komplexe dynamische Zugriffsregeln, wie sie beispielsweise bei RBAC (Role-based-access-control)[2] oder ABAC (Attribute-based-access-control)[3] nötig sind, unabhängig von Anwendungen definieren und verwalten zu können, hat sich der OASIS-Standard XACML (eXtensible Access Control Markup Language)[4] etabliert, welcher eine herstellerunabhängige Sprache - basierend auf XML - zur Verfügung stellt (Stepien et al 2014).

Unsere Analyse der bisherigen Arbeiten konnte aufzeigen, dass es im Rahmen komplexer Policy-Management Systeme erheblichen Forschungsbedarf bzgl. einer geeigneten Nutzerzentrierung gibt. Speziell die Administration und die Verwaltung von Policies ist auch für erfahrene Nutzer ein fehleranfälliger und oft langwieriger Prozess, welcher oft zu inkorrekten Access-Control-Policies führt und damit ggf. zu unberechtigtem Zugriff (Nergaard et al. 2015; Stepien et al. 2014; Church & Whitten 2009). Anwender haben ebenso Schwierigkeiten, Regeln ohne zusätzliche Unterstützung von IT-Personal auszudrücken (Sánchez et al. 2006). Oft sind hierbei umfangreiche Schulungen und/oder zusätzliches Personal nötig, um Aufgaben im Rahmen des Policy-Managements ausführen zu können (Beckerle & Martucci 2013; Reeder et al. 2007).

3 Gestaltungsprinzipien

In der Literatur können verschiedene Prinzipien zur Gestaltung von Usable-Security-Systemen gefunden werden. Der folgende Abschnitt stellt kompakt die wesentlichsten Arbeiten als Orientierungsinstrument für andere Wissenschaftler zur Verfügung.

---

2 Weitere Informationen sind verfügbar unter: http://csrc.nist.gov/groups/SNS/rbac.

3 Weitere Informationen sind verfügbar unter: http://csrc.nist.gov/projects/abac.

4 Weitere Informationen sind verfügbar unter: https://www.oasis-open.org/standards

Eine der ersten Gestaltungsprinzipien erarbeiteten Saltzer und Schroeder. Die im Jahr 1975 entstandene Arbeit enthält die nachfolgend gelisteten allgemeinen Prinzipien zur Gestaltung von Computersystemen zum Schutz von Informationen (Smith 2012).

1. *„Economy of mechanism“*: Das System muss wirksam und einfach sein.
2. *„Fail-safe defaults“*: Zugriff wird nur nach Erlaubnis erteilt.
3. *„Complete mediation“*: Alle Zugriffsaktivitäten werden geprüft.
4. *„Open design“*: Die eingesetzten Verfahren und Mechanismen werden offengelegt.
5. *„Separation of privilege“*: Die Vergabe von Rechten darf nicht nur auf einer Bedingung beruhen.
6. *„Least privilege”*: Es sollten nur so viele Rechte definiert werden wie nötig.
7. *„Least common mechanism”*: Potentielle Informationsflüsse zwischen Anwendern sollten minimiert werden.
8. *“Psychological acceptability”*: Das mentale Model des Anwenders passt zu der Gestaltung des Systems.

Die Bedeutung der Arbeit von Saltzer und Schroeder hat mit zunehmender Verbreitung und Vielfältigkeit der Anwender im Laufe der Zeit zugenommen. Computersysteme hielten Einzug in den Arbeitsalltag und auch in den privaten Bereich der verschiedensten Anwendergruppen. 1996 haben Zurko und Simon die Nutzerzentrierung durch die Prägung des Begriffes „User-Centered-Security“ weiter in den Fokus der Gestaltung von Security-Systemen gesetzt (Zurko & Simon 1996). Erstmals wurde dadurch Usability als primäres Ziel und Motivation bei der Gestaltung von Security-Modellen, Mechanismen, Systemen sowie Software definiert. Um kritische Merkmale im Bereich von Security, wie bspw. Demotivation, unzureichende Abstraktionsfähigkeit von Anwendern und wenig System-Feedback entgegenzuwirken, haben verschiedene Forscher Prinzipien zum Design von Usable-Security-Systemen entwickelt. Ein Auszug der relevantesten Arbeiten ist in Tabelle 1 dargestellt.

Quelle	**Prinzipien**
„Why Johnny can't encrypt: a usability evaluation of PGP 5.0” - Properties of the usability problem for security (Whitten & Tygar 1999)	The unmotivated user property, The abstraction property, The lack of feedback property, The barn door property, The weakest link property
„User Interaction Design for Secure Systems“ (Yee 2002)	Path of least resistance, Appropriate boundaries, Explicit authorization, Visibility, Revocability, Expected ability, Trusted path, Identifiability, Expressiveness, Clarity
„A Framework for secure Computer Interaction” - Criteria for a successful HCI (Johnston 2004)	Visibility of system status, Aesthetic and minimalist design, Satisfaction, Convey features, Learnability, Trust

„Human-Computer Interaction Opportunities for Improving Security/Privacy“ - Possible Strategies for usable security interfaces (Shneiderman 2005)	Reduce complexity, Cleaner cognitive model, Show consequences of decisions, Show dynamics of activity with viewable log
„Design Principles and Patterns for Computer Systems That Are Simultaneously Secure and Usable”- General principles for aligning security and usability (Garfinkel 2007)	Least surprise, Good security now, Standardized security policies, Consistent, meaningful vocabulary, Consistent controls and placement, No external burden

Tabelle 1: Auszug von Prinzipien für die Gestaltung von Usable-Security-Systemen

In weiteren Arbeiten streben wir die Kategorisierung und Abgrenzung der dargestellten Prinzipien sowie einen Abgleich dieser teilweise spezifischen Prinzipien mit allgemeinen Gestaltungsprinzipien im Bereich der Mensch-Maschine-Interaktion (z.B. DIN EN ISO 9241 - 110)[5] an. Durch diese Vorgehensweise könnte es möglich werden ein konsolidiertes Grundgerüst von Gestaltungsprinzipien für sichere und gebrauchstaugliche Systeme zu erarbeiten.

4 Usable-Policy-Management

Nachfolgend werden relevante Arbeiten im betrachteten Spannungsfeld kompakt vorgestellt. Anschließend erfolgen die Darstellung der eingesetzten Strategien, die Nennung von Herausforderungen, die Ableitung von Forschungsfragen und die Darstellung eines möglichen Lösungsansatzes, zur Erlangung eines Usable-Policy-Managements in kollaborativen Arbeitsumgebungen.

4.1 Aktuelle Arbeiten

Eine der ersten Arbeiten im Bereich des Usable-Policy-Managements lieferten 1997 Zurko et al. mit „Adage“. „Adage“ umfasst eine interaktive graphische Benutzeroberfläche („Visual Query Builder“) für die Erstellung von RBAC-Polices (Zurko 1999). Es werden Building Blocks und Templates zur Erstellung eingesetzt, um den Nutzer zu unterstützen.1999 stellten Bartal et al. mit Firmato ein Firewall-Management-Toolkit vor, welches Firewall-Rules visualisiert. Hewlett Packard entwickelten 1999 Power („Policy-Wizard-Engine for Refinement“), ein integratives Authoring Werkzeug, welches erstmals die beiden Nutzerrollen der „Experten“ (Technical Knowledge) und „Berater“ (Business Knowledge) einführt (Mont et al. 1999). Experten generieren hierbei Policy-Templates, welche durch den Berater innerhalb einer GUI genutzt werden können. Mit SPARCL („Server Privacy ARchitecture and Capab-

5 Ergonomie der Mensch-System-Interaktion - Teil 110: Grundsätze der Dialoggestaltung.

iLity Enablement“), einem Policy-Management Tool aus dem Jahre 2005, können Policies entweder in natürlicher Sprache oder durch strukturierte Eingabe / Auswahl (bspw. durch Listboxen) von Elementen und Regeln erzeugt werden (Brodie et al. 2005). Reeder et al. haben sich auf die Visualisierung des Zustandes von Policies spezialisiert und lieferten 2007 mit „Expandable Grids“ eine Matrixvisualisierung, welche effizient die „High-Level-Zugriffsrechte“ (wie „Allow, Deny, Some access allowed“) eines Users oder einer Gruppe von Usern auf Ressourcen visualisiert (Reeder et al. 2011). Stepien, Matwin und Felty erarbeiteten 2011 eine nicht technische Notation für XACML-Policies, welche 2014 weiter bearbeitet wurde (Stepien et al. 2014, Stepien et al. 2011). In dieser können Benutzer durch Checkboxen und Listboxen strukturiert Policies innerhalb einer GUI erstellen. Um dies zu ermöglichen, muss im Vorfeld das Datenmodell von Experten erstellt werden. Johnson et al. haben 2011 einen Prototypen zur iterativen Policy-Erstellung vorgestellt (Johnson et al. 2010). In dieser Arbeit wird davon ausgegangen, dass mindestens drei Nutzergruppen an diesem Prozess beteiligt sind, nämlich Policy-Authors, Template-Authors und Policy-Element-Authors (definiert spezifische Policy Domainelemente). Auch in diesem Ansatz werden Templates eingesetzt. Nergaard et al. entwickelten 2015 einen auf der visuellen Programmiersprache „SCRATCH“ basierenden graphischen Policy-Editor für XACML, welcher sich an dem vom MIT entwickelten SCRATCH-Editor orientiert (Nergaard et al. 2015). Dieser Editor nutzt Building Blocks, welche verwendet werden, um visuell eine XACML-Policy zu erstellen. Es wird auch hier technisches Know-how vorausgesetzt, da die Funktionen und Syntax von XACML erhalten bleiben.

4.2 Strategien und Herausforderungen

Eine Untersuchung der im vorherigen Abschnitt vorgestellten Arbeiten lässt die folgenden Strategien zur Optimierung der Mensch-Maschine-Schnittstelle erkennen: *Natürlichsprachliche Texteingabe*, *alternative Syntax*, *Templates*, *strukturierte Eingaben*, *Reduzierung* des *Funktionsumfangs* und *visuelle Unterstützung*.

Wird der betrachtete Nutzungskontext der aktuellen Arbeiten analysiert, so wird erkennbar, dass die bisherigen Arbeiten unzureichend Charakteristiken von kollaborativen Arbeitsumgebungen beachtet haben. Obwohl generell gute Ansätze erkennbar sind, so fehlt dennoch eine ganzheitliche nutzerzentrierte Betrachtung, wie Policy-Management anwenderfreundlich und agil auf Administrations- und Anwendungsebene gestaltet werden kann. Viele der betrachteten Ansätze und Lösungen setzen zudem entweder Programmierkenntnisse oder technisches Verständnis über die Syntax und die Elemente einer Policy voraus. Gleichermaßen wird deutlich, dass das Zusammenspiel zwischen den einzelnen Akteuren meist auf Annahmen statt auf Nutzeranalysen (z.B.: Interviews, Beobachtungen, On-Site Visits) beruht. Um diese Lücken zu schließen, muss analysiert werden, wie und welche Nutzer aktuell im Bereich des Policy-Managements innerhalb kollaborativer Arbeitsumgebungen zusammenarbeiten, welche Anforderungen sich daraus ableiten lassen und wie diese durch zukünftige Komponenten und Lösungen anwenderfreundlich gestaltet werden können. Aus dem beschriebenen Sachverhalt können u.a. die folgenden *Forschungsfragen* abgeleitet werden:

1. Welche *Anforderungen* sind für ein gebrauchstaugliches Policy-Management innerhalb kollaborativer Arbeitsumgebungen für die verschiedenen Akteure nötig?

2. Wie können existierende Policy-Management-Komponenten und Lösungen im Rahmen kollaborativer Zusammenarbeit *erweitert* und *verbessert* werden, so dass die Gebrauchstauglichkeit der beteiligten Akteure verbessert wird?

3. Wie kann ein *ganzheitliches* nutzerzentriertes (mikro- und makroskopisches) Policy-Management innerhalb kollaborativer Arbeitsumgebungen gestaltet werden?

4.3 Lösungsansatz

Basierend auf unseren gewonnenen Erkenntnissen stellen wir fest, dass bei den aktuellen verfügbaren Lösungen im Rahmen von kollaborativen Policy-Management signifikante Usability-Optimierungspotenziale vorhanden sind. Zur Bearbeitung und Optimierung der genannten Problemfelder stellen wir die folgende *Arbeitshypothese* vor.

Die Berücksichtigung von etablierten Gestaltungsprinzipien (siehe Kapitel 3), die Anwendung einer nutzerzentrierten Vorgehensweise sowie die Übertragung des aus den Sozialwissenschaften bekannten Konzeptes von „*Structure and Agency*“[6], könnte vielsprechende Ergebnisse zur Verbesserung der Gebrauchstauglichkeit im Rahmen des Policy-Managements liefern. Die Struktur könnte die Möglichkeit der Zusammenarbeit zwischen den Agenten vorgeben, wobei sich Struktur und Agenten gegenseitig beeinflussen können (Barker 2005; Karp 1986). Über die Agenten (bspw.: Personen, Software, Hardware, Prozesse oder auch Policies) kann wiederum die Struktur beeinflusst werden. Denn nur, wenn eine bestimmte Struktur in einem bestimmten Zustand vorhanden ist, dürfen bestimmte Handlungen vollzogen werden. Eine Modellierung der Struktur könnte beispielsweise mittels der Web Ontology Sprache (OWL) in Form einer Ontologie erfolgen. Individuen wie Personen und Gruppen entsprechen in diesem Fall Knoten innerhalb eines „sozialen Graphen“, Beziehungen den Kanten. In Regeln könnte dann festgehalten werden, welche relationalen Bedingungen für Subjekt und Ressource gelten müssen, damit eine bestimmte Aktion erlaubt wird. Diese Regeln könnten mittels einer regelbasierten Sprache formuliert werden. Die Regeln könnten dann wiederum zu komplexeren Policies bzw. Regelwerken (z.B. in XACML) zusammengefasst werden. Bei Zugriffskontrollen nach diesem Modell könnten dann die Handlungsmöglichkeiten („Agency“) eines Subjekts aus den Strukturen („Structure“) des „sozialen Graphen“ abgeleitet werden, indem Attributwerte aus dem Graphen extrahiert werden. Die „Agency“ eines Agenten (also eines Zugriffssubjekts) ergibt sich dabei aus der Menge der erlaubten Handlungen, die dieses Subjekt zu seinem aktuellen Zeitpunkt gegenüber anderen Agenten (Ressourcen) ausüben darf. Diese erlaubten Handlungen ergeben sich aus den Strukturen, über die das Subjekt mit anderen Agenten verbunden ist, also aus der Position des Subjekts im „sozialen Graphen“. Es ist davon auszugehen, dass bestimmte Konstellationen der Struktur (Strukturmuster) für die Vergabe von Zugriffsrichtlinien Bedeutung haben und häufig Anwendung finden (beispielsweise binäre Strukturen sowie Gruppen- und Rollenstrukturen). Wenn es möglich wird, die organisatorische Struktur als Ontologie abzubil-

6 Die Kernfrage ist, ob und wie die soziale Struktur das menschliche Handeln beeinflusst und ob oder wie durch menschliches Handeln soziale Strukturen geändert und erzeugt werden.

den sowie basierend auf dieser typische Strukturmuster abzuleiten, so könnte anschließend ein grafisches Nutzerinterface dazu beitragen, dass Policies effizienter erstellt werden können, da nur eine Parametrierung der zu erfüllenden Konstellation (Zustand) der „sozialen Struktur“ vollzogen werden müsste.

5 Zusammenfassung und Ausblick

Der Bedarf für ein gebrauchstaugliches Policy-Management, speziell in kollaborativen Umgebungen, ist vorhanden und wird in Zukunft steigen. Wir gehen davon aus, dass ein Ansatz bestehend aus einer Kombination der folgenden drei Aspekte vielversprechend ist.

- Integration von Usability-Engineering-Aktivitäten und Usable-Security-Gestaltungsempfehlungen
- Betrachtung von Gruppenprozessen innerhalb von Unternehmen zur Beachtung von spezifischen Interaktionscharakteristiken (wie Agilität und Flexibilität)
- Nutzung sozialer Modele und sozialer Modellierung von Akteuren und Ressourcen zur Reduzierung der Komplexität

Bei der Entwicklung von zukünftigen Lösungen könnte die Berücksichtigung der genannten Aspekte entscheidend für den Erfolg von sicheren und zugleich gebrauchstauglichen kollaborativen Arbeitsumgebungen werden.

Literaturverzeichnis

Barker, C. (2005). *Cultural Studies: Theory and Practice*. London: Sage.

Bartal, Y., Mayer, A., Nissim, K. & Wool, A. (1999). Firmato: Novel Firewall Management Toolkit. In In *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, Piscatawa: IEEE, S.17-31.

Beckerle, M. & Martucci, L. A. (2013*).* Formal Definitions for Usable Access Control Rule Sets - From Goals to Metrics*.* In *Proceedings on Ninth Symposium on Usable Privacy and Security.* Pittsburgh: ACM. S.2-11.

Brodie, C., Karat, C.-L & Karat, J. (2005). Usable Security and Privacy: A Case Study of Developing Privacy Management Tools. In *Proceedings of the 2005 symposium on Usable privacy and security Pages*. Pittsburgh: ACM. S.35-43.

Church, L. & Whitten, A. (2009). Generative Usability: Security and User Centered Design beyond the Appliance, In *Proceedings of the 2009 workshop on New security paradigms workshop Pages*. Oxford: ACM. S.51-58.

Garfinkel, S. L. (2005*). Design Principles and Patterns for Computer Systems That Are Simultaneously Secure and Usable*, Cambridge: Massachusetts Institute of Technology.

Johnson, M. (2012). *Toward Usable Access Control for End-users: A Case Study of Facebook Privacy Settings*, New York: Columbia University.

Johnson, M., Karat, J., Karat, C.M. & Grueneberg, K. (2010). Usable Policy Template Authoring for Iterative Policy Refinement. In *Proceedings of the IEEE International Symposium on Policies for Distributed Systems and Networks*, New York: Columbia University. S.18-21.

Johnston, J. (2004). *A Framework for secure Computer Interaction*, Johannesburg: Rand Afrikaans University, S.21-22.

Karp, I. (1986). Agency and social theory: a review of Anthony Giddens. In *American Ethnologist*, 13(1), Oxford: Blackwell Publishing. S.131-137.

Mont, M. C., Baldwin, A. & Goh, C. (1999). *POWER Prototype: Towards Integrated Policy-Based Management*, Bristol: Hewlett Packard.

Nergaard, H., Ulltveit-Moe & N., Gjøsæter, T. (2015). A Scratch-based Graphical Policy Editor for XACML, In *Proceedings of the 1st International Conference on Information Systems Security and Privacy*, Berlin: Springer Verlag. S.182-191.

Reeder, R. W., Bauer, L., Cranor, L. F., Reiter, M. K. & Vaniea, K. (2011). More than Skin Deep: Measuring Effects of the Underlying Model on Access-Control System Usability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Oxford: ACM. S. 2065-2074.

Reeder, R. W., Karat, C.-M. , Karat, J. & Brodie, C. (2007). Usability Challenges in Security and Privacy. In *Human-Computer Interaction - INTERACT 2007*. Berlin: Springer Verlag. S.141-155.

Sánchez, M., López, G., Gomez-Skarmeta, A.F. & Canovas, O. (2006). Using Microsoft Office InfoPath to Generate XACML Policies, In *Proceedings of the International Conference on Security and Cryptography*, Lissabon: INSTICC Press, S. 379-386.

Shneiderman, B. (2005). *Human-Computer Interaction Opportunities for Improving Security/Privacy*, College Park: University of Maryland, S. 34.

Smith, R.E. (2012). A Contemporary Look at Saltzer and Schroeder's 1975 Design Principles, In *Security & Privacy*, vol.10, no.6, Piscatawa: IEEE, S. 20-25.

Stepien, B., Felty, A. & Matwin, S. (2014). A non-technical XACML target editor for dynamic access control systems, In *Collaboration Technologies and Systems*, Piscatawa: IEEE, S.150-157.

Stepien, B., Matwin, S. & Felty, A. (2011). Advantages of a Non-Technical XACML Notation in Role-Based Models, In *Proceedings of the Ninth Annual International Conference on Privacy, Security and Trust*. Piscatawa: IEEE, S. 193-200.

Whitten, A. & Tygar, J. D. (1999). Why Johnny can't encrypt: a usability evaluation of PGP 5.0, In *Proceedings of the 8th USENIX Security Symposium*. Berkeley: USENIX Association, S.169-184.

Yee, K.-P. (2002). User Interaction Design for Secure Systems, In *Proceedings of the 4th International Conference on Information and Communications Security*, London: Springer-Verlag, S.278-290.

Zurko, M. E., Simon, R. T. & Sanfilippo, T. (1999). A user-centered, modular authorization service built on an RBAC foundation. In *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, Piscatawa: IEEE, S.57-71.

Zurko, M. E. & Simon, R. T. (1996). User-centered security, In *Proceedings of the 1996 workshop on New security paradigms*, New York: ACM, S.27-33.

Kontaktinformationen

Sascha Wagner (sascha.wagner@fernuni-hagen.de)
Dominic Heutelbeck (dheutelbeck@ftk.de)