Credential-basierte Zugriffskontrolle: Wurzeln und ein Ausblick

Joachim Biskup

biskup@ls6.informatik.uni-dortmund.de

Abstract: Verfahren der Credential-basierten Zugriffskontrolle liefern einen wichtigen Beitrag zur Sicherheit in offenen, interoperablen IT-Systemen. Wir zeigen, dass die Wurzeln der Credential-basierten Zugriffskontrolle in der Zusammenführung mehrerer Entwicklungslinien der Informatik liegen. Als Ausblick wird ein allgemeines Modell für die Credential-basierte Zugriffskontrolle umrissen.

1 Einleitung

Sicherheit muss durch aufeinander bezogene Techniken der Informatik einerseits und organisatorische Maßnahmen in der jeweiligen Anwendungsumgebung andererseits erreicht werden. Zu den Techniken der Informatik gehören die Zugriffskontrolle und Überwachung sowie die Kryptographie. Diese Techniken ergänzen und stützen sich wechselseitig.

Herkömmliche Verfahren zur Zugriffskontrolle sind ursprünglich für zentrale Systeme entwickelt worden. Offene, interoperable IT-Systeme erfüllen nicht mehr die für die herkömmliche Zugriffskontrolle benutzten Voraussetzungen. Credential-basierte Verfahren zur Zugriffskontrolle versprechen, für die neuen Herausforderungen tragfähige Lösungen zu liefern. Insbesondere können mit Hilfe von Credentials Autorisierungen nicht nur an Identitäten, sondern auch an andere Merkmale von Systemteilnehmern gebunden werden. Und Credentials können aufgrund ihrer kryptographisch gesicherten Beglaubigungen auch über unsichere Kanäle bei entfernten Komponenten genutzt werden.

2 Wurzeln

Verfahren der Credential-basierten Zugriffskontrolle verbinden die Ergebnisse mehrerer Entwicklungslinien der Informatik, die im Folgenden kurz skizziert werden.

Zugriffskontrolle

Herkömmliche Verfahren zur Zugriffskontrolle [SC01] in zentralen Systemen setzen voraus, dass an im Vorhinein bekannte Identitäten wohlbestimmte Zugriffsrechte für spezifische Zugriffe zu Diensten oder Daten vergeben werden. Ferner wird ein sicherer lokaler

Kanal angenommen, über den solche Zugriffe angefordert werden. Die Zugriffskontrolle überprüft dann die Authentizität der Anforderung und entscheidet über Erlaubnis oder Verbot aufgrund der vergebenen Zugriffsrechte. Offene, interoperable IT-Systeme erfüllen nicht mehr die Voraussetzungen herkömmlicher Verfahren zur Zugriffskontrolle, die somit nur noch eingeschränkt oder geeignet angepasst einsetzbar sind.

Capabilities

Für die herkömmliche Zugriffskontrolle wird bereits frühzeitig das Konzept von Capabilites vorgeschlagen und entwickelt [Fa74, Wu81]: Wenn konzeptionell einem Teilnehmer (Subjekt) s bezüglich einem Dienst (Objekt) o eine Handlung (Aktion) a erlaubt wird, so kann solch eine Erlaubnis dadurch implementiert werden, dass dem Teilnehmer durch eine Capability der abstrakten Form [o,a] eine entsprechende Bescheinigung ausgestellt wird. Dieser Ansatz wird später weiterentwickelt im Hinblick auf verteilte Systeme, zum Beispiel für verteilte Betriebssysteme wie Amoeba [Ta90] oder für Client-Server Systeme wie Kerberos [Mi87]. In konkreter Form müssen Capabilities gegen Fälschungen und Missbrauch geschützt werden. Während man solch einen Schutz in zentralen Systemen noch allein durch eine auf einen Sicherheitskern abgestützte Systemarchitektur anstrebt, erfordern verteilte Systeme den Einsatz von zusätzlichen kryptographischen Maßnahmen.

Authentisierungen

Authentisierungen und die entsprechenden Überprüfungen sollen sicherstellen, dass bei einer Anforderung von Diensten der vorgeblich anfordernde Teilnehmer der tatsächlich Anfordernde ist und als solcher korrekt erkannt werden kann. Insbesondere muss ein Anfordernder, der mit der Anforderung eine Capability vorweist, als deren rechtmäßiger Besitzer erkannt werden. In zentralen Systemen werden diese Anforderungen meistens auf Passworten oder anderen Merkmalen abgestützt, die jeweils für einen einzelnen menschlichen Teilnehmer als eigentümlich und eindeutig zuordenbar angesehen werden. In verteilten Systemen kann man die digitale Darstellung solcher Merkmale im Allgemeinen nicht ohne weiteres (fälschungs)sicher übertragen, und man muss deshalb zusätzliche Sicherheitsmaßnahmen, vornehmlich kryptographischer Art, einsetzen.

Symmetrische kryptographische Authentisierungen

Benutzt man symmetrische Kryptographie, so muss zwischen den Beteiligten im Vorhinein ein Geheimnis als symmetrischer Schlüssel festgelegt werden. Dieser Schlüssel wird dann einerseits bei der Erzeugung von Message Authentication Codes (MACs) bzw. von Challenges eingesetzt und andererseits bei deren Überprüfung bzw. für erwidernde Responses genutzt. Die Festlegung eines Schlüssels zwischen verteilt arbeitenden und zuvor einander unbekannten Teilnehmern bedarf jeweils eines als vertrauenswürdig angesehenen Dritten. Beispiele hierfür sind das Needham-Schroeder-Protokoll [NS78] und seine Varianten, die beispielsweise auch in Kerberos eingesetzt werden, und Varianten des Diffie-Hellman-Protokolls [Di92].

Digitale Unterschriften, Zero-Knowledge Proofs und Zertifikate

Benutzt man asymmetrische Kryptographie, so benötigt man als vertrauenswürdig angesehene Dritte zur Zertifizierung öffentlicher Schlüssel. Durch die Zertifizierung wird eine Bindung zwischen einem öffentlichen Schlüssel und dem rechtmäßigen Besitzer des zu-

gehörigen geheimen Schlüssels nachprüfbar beglaubigt. Solche Zertifizierungen werden auch hierarchisch durch eine sogenannte Public Key Infrastructure durchgeführt [ES00]. Ein Empfänger von Zertifikaten prüft jeweils eine ganze Kette von Zertifikaten, die von einem Dritten ausgehen muss, den der Empfänger aus eigenem Vermögen und Entschluss als vertrauenswürdig einschätzt. Ein weit verbreiteter Standard hierfür ist X.509 [Ie98]. Ein Teilnehmer kann dann seinen geheimen Schlüssel für Authentisierungen verwenden, insbesondere für digitale Unterschriften [DH76, Pf96] bezüglich Anforderungen von Diensten, Capabilities und Zertifikaten, oder auch im Rahmen von Protokollen für Zero-Knowledge Proofs [FS86, Go99]. Der jeweilige Partner nutzt den als zugehörig zertifizierten öffentlichen Schlüssel für die jeweils anstehenden Überprüfungen.

Credentials

Die asymmetrische Kryptographie eröffnet auch eine ganz neue Möglichkeit, Berechtigungen für die Inanspruchnahme von digitalen Diensten auszustellen, nämlich in Form von Credentials [Ch85, Br00]. Ein Credential ist ein digital unterschriebenes digitales Dokument, mit dem eine solche Berechtigung nicht mehr an eine herkömmliche Identität eines Teilnehmers, sondern an einen öffentlichen Schlüssel gebunden wird. Dadurch kann auch gegebenenfalls eine Berechtigung zunächst anonym in Anspruch genommen werden. Trifft man geeignete Vorsorgemaßnahmen, so kann die Anonymität bei einem Missbrauch und auch nur dann wieder aufgedeckt werden. Credentials können für verschiedene Arten von Berechtigungen ausgegeben werden. Man unterscheidet insbesondere einmal und mehrmals verwendbare sowie persönliche und weitergebbare. Zum Beispiel sollen elektronische Münzen nur einmal verwendbar und weitergebbar sein, aber Capabilities im Allgemeinen mehrfach verwendbar und persönlich. Darüber hinaus können Credentials auch im Hinblick auf beliebige „freie Eigenschaften“ des vorgesehenen Halters ausgestellt werden: Sie bestätigen dann, dass der Halter über die betreffende Eigenschaft verfügt.

Middleware-Systeme

Middleware-Systeme wie Corba [Om01] oder ähnliche Ansätze für verteilte Systeme [TS02] stellen die Hilfsmittel bereit, um weitgehend autonome Komponenten auf transparente Weise zu offenen und interoperablen Gesamtsystemen zusammenzufassen. Innerhalb eines solchen Gesamtsystems soll eine Komponente die Dienste anderer Komponenten auf sichere Weise anfordern können. Dafür wird die Vergabe von Berechtigungen und deren Authentisierungen und Überprüfungen mit Hilfe von „Sicherheitsattributen“ geregelt, die man als eine geeignete Art von Credentials deuten kann. Eine besondere Fragestellung ergibt sich dabei hinsichtlich der Delegation von Berechtigungen.

Delegation und Simple Public Key Infrastructure

Die Vorschläge für SPKI/SDSI [El99, El01, Cl01], Simple Public Key Infrastructure, oder für ähnliche Systeme wie KeyNote [Bl99] führen einige der schon genannten Gesichtspunkte zusammen. Ein Betreiber von Diensten wird als deren Besitzer angesehen. Als Besitzer kann er die Inanspruchnahme der Dienste lokal mit Hilfe von Zugriffskontroll-Listen regeln. Die Begünstigten erhalten jeweils als Bescheinigung ein vom Besitzer digital unterschriebenes Autorisierungs-Zertifikat, also eine Art von Credential. Solch ein Autorisierungs-Zertifikat kann als delegierbar markiert werden, so dass dann ein Begünstigter wiederum entsprechende Autorisierungs-Zertifikate ausstellen kann. Fordert ein

direkt oder transitiv Begünstigter einen Dienst an, so überprüft der Besitzer die von ihm ausgehende Kette von Zertifikaten und versucht sie auf einen Eintrag in seiner lokalen Zugriffskontroll-Liste zu reduzieren.

3 Ausblick: ein allgemeines Modell

Die Credential-basierte Zugriffskontrolle steht immer noch am Anfang ihrer Entwicklung [SC01], und es gibt auch kein Einvernehmen über ein allgemeines Modell. Im Folgenden wird ein Ansatz für ein solches Modell umrissen [Al02, BK02], wobei bezüglich der Literatur viele Bezeichnungen überladen gebraucht werden. Eine Instanz des zu entwickelnden Modells wird durch Abbildung 1 veranschaulicht. Das Modell besteht aus zwei, im Allgemeinen aufeinander bezogenen Teilen, die in konkreten Anwendungen gegebenenfalls auch nur rudimentär auftreten. Bei einer Ausgestaltung müssen neben informatorisch-technischen Fragestellungen auch organisatorische Aufgaben bewältigt werden.

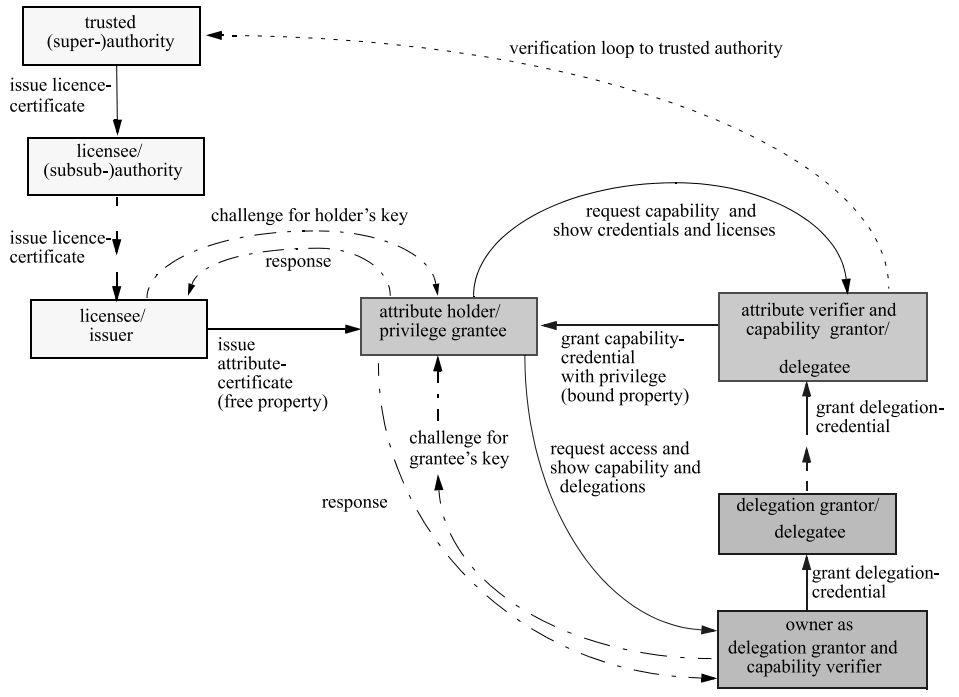

Abbildung 1: Eine Instanz des Modells

Ein erster Teil behandelt *Zertifikate*, die einem durch einen öffentlichen Schlüssel gekennzeichneten Teilnehmer gewisse *Attribute* (grundsätzlich jegliche beglaubigbare Eigenschaften oder Merkmale) zusprechen. Solche Attribute werden hier als *„frei“* benannt

in dem Sinne, dass deren spätere Verwendung noch nicht festgelegt worden ist. Solche Zertifikate werden hier *Attribut-Zertifikate* genannt. Hierzu gehören als Spezialfall auch Zertifikate, die einen öffentlichen Schlüssel an eine bürgerliche Identität binden. Die Aussteller solcher Attribut-Zertifikate benötigen dazu im Allgemeinen eine Erlaubnis, die wiederum durch Zertifikate dargestellt werden, die hier als *Lizenz-Zertifikate* benannt werden. Die Überprüfung eines Attribut-Zertifikats verlangt dann stets auch die Überprüfung einer geeigneten *Kette* von Lizenz-Zertifikaten.

Der zweite Teil behandelt die eigentlichen *Credentials*, die einem durch einen öffentlichen Schlüssel gekennzeichneten Teilnehmer gewisse *Autorisierungen* oder *Privilegien* für bestimmte fremde Dienste zusprechen. Solch eine Autorisierung oder solch ein Privileg wird hier als *„gebunden“* benannt in dem Sinne, dass deren spätere Verwendung bereits im Hinblick auf den jeweiligen Dienst festgelegt worden ist. Solche Credentials werden hier *Capability-Credentials* genannt. Sie werden vor allem vom Besitzer des betroffenen Dienstes vergeben. Der Besitzer kann jedoch die Vergabe auch an andere Teilnehmer delegieren. Die Deleganten benötigen dann eine Erlaubnis, um im Namen des Besitzers ihrerseits Capability-Credentials zu vergeben. Diese Erlaubnis wird wiederum durch Credentials dargestellt, die hier als *Delegation-Credentials* benannt werden. Die Überprüfung eines Capability-Credentials verlangt stets auch die Überprüfung einer geeigneten *Kette* von Delegation-Credentials. Solche Überprüfungen werden zum Zeitpunkt der Anforderung des Dienstes durch den Besitzer des Dienstes vorgenommen.

Die beiden Teile (einer Instanz des Modells) überschneiden sich jeweils in zwei Teilnehmern: Der eine Teilnehmer bemüht sich als *Attribut-Halter* unter Vorlage von entsprechenden Attribut-Zertifikaten um den Erwerb von Capability-Credentials, d.h., er möchte der *Begünstigte* im Hinblick auf die der Capability entsprechenden Dienste werden. Der andere Teilnehmer tritt als *Vergeber* von Capability-Credentials auf, d.h., er vergibt Autorisierungen oder Privilegien entsprechend einer Sicherheitspolitik, die sich auf Attribute abstützt. Dazu muss er insbesondere als *Attribut-Überprüfer* handeln. Vereinfacht gesprochen beinhaltet die Interaktion der beiden Teilnehmer die Umwandlung „freier Attribute“ in „gebundene Privilegien“. Im einfachsten Fall besteht das gesamte Modell nur aus solchen Teilnehmern.

Literaturverzeichnis

[Al02] Altenschmidt, C., Biskup, J., Flegel, U., Karabulut, Y., Secure mediation: requirements, design and architecture, Journal of Computer Security, to appear.

[BK02] Biskup, J., Karabulut, Y., A hybrid PKI model with an application for secure mediation, Proc. 16th Annual IFIP WG 11.3 Working Conference, Cambridge, UK, July 2002, to appear.

[Br00] Brands, S.A., Rethinking Public Key Infrastructures and Digital Certificates, MIT Press, Cambridge-London, 2000.

[Bl99] Blaze, M., Feigenbaum, J., Ioannidis, J., Keromytis, A., The KeyNote trust management system version 2, RFC 2704, IETF, 1999.

[Ch85] Chaum, D., Security without identification: transaction systems to make big brother obsolete, Comm. of the ACM 28,10 (1985), pp. 1030-1044.

[Cl01] Clarke, D., Elien, J.-E., Ellison, C., Fredette, M., Morcos, A., Rivest, R.L., Certificate chain discovery in SPKI/SDSI, Journal of Computer Security 9 (2001), pp. 285-322.

[DH76] Diffie, W., Hellman, M.E., New directions on cryptography, IEEE Transactions on Information Theory 22,6 (1976), pp. 644-654.

[Di92] Diffie, W., Van Oorschot, P.C., Wiener, M.J., Authentication and authenticated key exchanges, Design, Codes and Cryptography 2 (1992), pp. 107-125.

[El99] Ellison, C.M., Frantz, B., Lampson, B., Rivest, R., Thomas, B.M., Ylonen, T., Simple public key certification, http://www.ietf.org/ids.by.wg/spki.html, 1999.

[ES00] Ellison, C.M., Schneier, B., Ten risks of PKI: what you're not being told about public key infrastructure, Computer Security Journal 16,1 (2000), pp. 1-7.

[El01] Ellison, C.M., SPKI/SDSI certificates, http://world.std.com/ cme/html/spki.html, 2001.

[Fa74] Fabry, R., Capability-based addressing, Comm. of the ACM 17,7 (1974), pp. 403-412.

[FS86] Fiat, A., Shamir, A., How to prove yourself: practical solutions to identification and signature problems, Proc. CRYPTO 86, LNCS 263, Springer, Berlin etc., 1986, pp. 186-194.

[Go99] Goldreich, O., Modern Cryptography, Probabilistic Proofs and Pseudorandomness, Springer, Berlin etc., 1999.

[Ie98] IETF X.509 Working Group, Public-key infrastructure (X.509), http://www.ietf.org/html.charters/pkix-charter.html, 1998.

[Mi87] Miller, S.P., Neuman, B.C., Schiller, J.I., Saltzer, J.H., Section E.2.1: Kerberos authentification and authorization system, M.I.T. Project Athena, Technical Report, Cambridge, Mass., 1987.

[NS78] Needham, R.M., Schroeder, M.D., Using encryption for authentification in large networks of computers, Comm. of the ACM 21,12 (1978), pp. 993-999.

[Om01] Object Management Group (OMG), Common Secure Interoperability, OMG Dokument ptc/2001-03-02, 2001, http://www.omg.org/cgi-bin/doc?ptc/2001-03-02.

[Pf96] Pfitzmann, B., Digital Signature Schemes - General Framework and Fail-Stop Signatures, LNCS 1100, Springer, Berlin etc., 1996.

[SC01] Samarati, P., de Capitani di Vimercati, S., Access control: policies, models, and mechanisms, FOSAD 2000 (Focardi, R., Gorrieri, R., eds), LNCS 2171, Springer, Berlin etc., 2001, pp. 137-196.

[Ta90] Tanenbaum, A.S., van Renesse, R., van Staveren, H., Sharp, G., Mullender, S., Jansen, J., van Rossum, G., Experiences with the Amoeba distributed operating system, Comm. of the ACM 33,12 (1990), pp. 46-63.

[TS02] Tanenbaum, A.S., van Steen, M., Distributed Systems, Prentice-Hall, Upper Saddle River, N.J., 2002.

[Wu81] Wulf, W.A., Levin, S.P., Harbison, S.P., Hydra/Cmmp: An Experimental Computer System, MacGraw-Hill, New York, 1981.