

Berichte aus der Kommunikationstechnik herausgegeben von
Prof. Firoz Kaderali

Band 6

Dirk Westhoff

**Ein dezentrales Agentensystem unter
Berücksichtigung von mehrseitiger Sicherheit**

Shaker Verlag
Aachen 2000

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Westhoff, Dirk:

Ein dezentrales Agentensystem unter Berücksichtigung von
mehrseitiger Sicherheit / Dirk Westhoff.

Aachen : Shaker, 2000

(Berichte aus der Kommunikationstechnik herausgegeben
von Prof. Firoz Kaderali ; Bd. 6)

Zugl.: Hagen, Univ., Diss., 2000

ISBN 3-8265-8117-2

Copyright Shaker Verlag 2000

Alle Rechte, auch das des auszugsweisen Nachdruckes, der auszugsweisen
oder vollständigen Wiedergabe, der Speicherung in Datenverarbeitungs-
anlagen und der Übersetzung, vorbehalten.

Printed in Germany.

ISBN 3-8265-8117-2

ISSN 1437-7497

Shaker Verlag GmbH • Postfach 1290 • 52013 Aachen

Telefon: 02407 / 95 96 - 0 • Telefax: 02407 / 95 96 - 9

Internet: www.shaker.de • eMail: info@shaker.de

Kapitel 6

Abschließende Betrachtungen

6.1 Zusammenfassung

Die vorliegende Arbeit vergleicht kommerzielle Agentensysteme sowie Agentensysteme, die aus der Forschung verschiedener Universitäten hervorgegangen sind. Alle Agentensysteme legen zentrale Strukturen zugrunde, die das System in einem konsistenten Zustand halten.

Stimmen die funktionalen Eigenschaften der vorgestellten Systeme weitgehend überein, so sind die Unterschiede hinsichtlich der Berücksichtigung und Realisierung von Verfahren und Protokollen zum Schutz einzelner Instanzen des Agentensystems signifikant. So bieten lediglich die Systeme IBM Aglets und D'Agents Ansätze zum Schutz des Rechners und der Arbeitsumgebung vor böswilligen Agenten. Beide Systeme bieten einen präventiven Schutz vor aktiven Angriffen, indem sie zur Laufzeit eines Agenten potentiell böswillige Operationen des Agenten unterbinden.

Dagegen sind in keinem der betrachteten Systeme Verfahren oder Protokolle integriert, die den Agenten vor passiven oder aktiven Angriffen böswilliger Arbeitsumgebungen schützen. Für das System Mole existiert lediglich ein Konzept, das den Agenten zu seiner Ausführungszeit gegen Leseangriffe schützen soll. Weitere Konzepte zum Schutz des Agenten auf seiner Reise sind in keinem der vorgestellten Agentensysteme berücksichtigt.

Diese Arbeit hatte es zum Ziel, neben bekannten funktionalen Eigenschaften, ein Agentensystem mit einer vollständig dezentralen Verwaltung zu erstellen. Gleichzeitig gehen in das System Aspekte mehrseitiger Sicherheit ein. So ist zum einen

der Schutz der Arbeitsumgebung und des Rechners vor böswilligen oder fehlerhaft programmierten Agenten gewährleistet. Zum anderen wurden Verfahren und Protokolle eingebunden, die Teile des Agenten präventiv vor passiven Angriffen einer Arbeitsumgebung schützen. Soweit aktive Angriffe einer Arbeitsumgebung nicht verhinderbar sind, sind sie nachträglich zumindest erkennbar und, in Fällen funktionsmodifizierender Angriffe auf den Binärcode des Agenten, zweifelsfrei nachweisbar.

Als Kommunikationsmechanismen sieht das vorgestellte Agentensystem die direkte und die indirekte Kommunikation zwischen Agenten vor. Zwei Agenten kommunizieren global direkt, indem der Client-Agent den Quellkontext des Server-Agenten in der Phase des Verbindungsaufbaus einbezieht. Die anschließende eigentliche Kommunikation erfolgt ausschließlich zwischen Client- und Server-Agenten. Erfolgt die direkte Kommunikation zwischen zwei Agenten lokal, erübrigt sich ein Verbindungsaufbau über den Quellkontext. Die indirekte Kommunikation zweier Agenten nutzt die Vermittlungskomponente Contact mit anschließender lokaler oder globaler direkter Kommunikation.

Im Kern des Agentensystems wird die dezentrale Verwaltung durch den Einsatz von Systemagenten erreicht. Jeder Kontext des Kerns verfügt über solche Systemagenten. Gegenüber herkömmlichen dezentralen Ansätzen netzbasierter verteilter Anwendungen bietet der Einsatz von Systemagenten den Vorteil, daß ein sich anmeldender Kontext keine Kenntnis über alle Internetadressen der gegenwärtig im System aktiven Arbeitskontexte besitzen muß. Die Kenntnis eines beliebigen aktiven Kontexts beim erstmaligen Starten eines weiteren Kontexts reicht aus.

Die Verwaltung der Schale des Agentensystems übernehmen deren Kontexte. Mit Hilfe des rückwärtigen Route-Durchlaufs teilen sie ihre Zustandsänderungen den eigenen reisenden Agenten mit.

Der Schutz des Kontexts und des Dateisystems eines Arbeitsrechners vor böswilligen oder fehlerhaft programmierten Agenten erfolgt über eine Zugriffs-/Zugangskontrolle und wird durch eine Anpassung des Security-Managers von Java erreicht. Es werden präventiv sicherheitskritische Operationen mobiler Agenten unterbunden, indem der Kontext zur Laufzeit des Agenten beobachtet, ob die gegenwärtig ausgeführte Aktion seinen Zugriffsrechten innerhalb einem lokal konfigurierten Verzeichnis von Rechte-Dateien entspricht.

Dieses Vorgehen macht das zweifelsfreie Authentifizieren von Agenten erforderlich. Nicht authentifizierbaren Agenten erlaubt der Kontext eine stark eingeschränkt-

te Funktionalität.

Der Schutz eines Agenten vor böswilligen Kontexten bedingt die Einteilung seiner Objekte in statische und dynamische Objekte.

Dynamische Objekte wie die Route des Agenten, seine mobilen Daten oder sein Logbuch unterliegen auf der Reise des Agenten Veränderungen. Derartige Objekte werden durch den Einsatz kryptographischer Verfahren partiell präventiv gegen passive Angriffe geschützt. Gleichzeitig gewährleisten die kryptographischen Verfahren einen repressiven Schutz der Objekte gegenüber aktiven Angriffen, indem sie solche Angriffe nachträglich erkennbar machen. Ein Qualitätsmerkmal von Verfahren zum repressiven Schutz dynamischer Daten ist der Zeitpunkt, zu dem der Angriff auf den Agenten erkennbar ist. Für Verfahren, die auf große zu schützende und u.U. redundante Datenmengen abzielen, tritt als ein weiteres Qualitätsmerkmal die anfallende Transportlast in den Vordergrund.

Zu den statischen Objekten eines Agenten zählen sein Profil sowie sein Binärkode. Der Schutz der statischen, auf der Reise des Agenten nicht veränderbaren Objekte ist repressiv und erfolgt durch den Einsatz kryptographischer Protokolle. Entwickelt wurden hierbei Protokolle, die zu den optimistischen Protokollen gehören. Protokolle dieser Klasse halten die Anzahl der erforderlichen Netztransaktionen im Falle einer ungestörten Arbeit klein. Für die zum Einsatz kommenden Protokolle erhöht sich erst im Falle eines tatsächlich erfolgten Angriffs auf den Agenten die Zahl der durchzuführenden Netztransaktionen. Dann wird ein vertrauenswürdiger Schiedskontext in das Protokoll einbezogen. In Abhängigkeit von dem zur Ausführung gelangenden Protokoll erzwingt der Schiedskontext das nochmalige Senden des korrekten Binärcodes, oder er stellt im Falle eines aktiven funktionsmodifizierenden Angriffs den Angreifer zweifelsfrei fest.

Alle Verfahren und Protokolle zum Schutz des Agenten halten Angriffen eines böswilligen Kontexts stand. Weiter werden sie vor dem Hintergrund koalierender böswilliger Kontexte diskutiert. Hier kann gezeigt werden, daß eines der vorgestellten Konzepte zum Schutz der Route auch koalierenden Angriffen einer Gruppe von böswilligen Kontexten standhält.

Da der Einsatz eines solchen Schutzkonzepts für die Route des Agenten u.a. verhindert, daß ein Kontext zur Laufzeit des Agenten entscheiden kann, welche seiner Koalitionspartner ebenfalls besucht werden sollen, können in diesem Zeitraum auch keinerlei aufschlußreiche Informationen zwischen den Koalitionspartnern ausgetauscht werden.

Diese Überlegungen betreffen nicht die Vorgänger- und die Nachfolgerstation eines böswilligen Arbeitskontexts. Beide Stationen können dem gegenwärtigen Arbeitskontext auch unter der Nutzung einer geschützten Route nicht verborgen bleiben. Ist der Nachfolger oder der Vorgänger eines böswilligen Kontexts ebenfalls böswillig, kann die Bildung eines koalierenden Angriffs mit diesen Parteien nicht verhindert werden. Aber auch in solchen Konstellationen ist das Bilden einer Koalition aus der Sicht böswilliger benachbarter Kontexte nicht immer sinnvoll. Sinnvolle Koalitionspartner für den aktuellen Arbeitskontext des Agenten sind nur solche Partner, die über ein weiterführendes protokollspezifisches Wissen verfügen oder laut Regel des Agentensystems verpflichtet wären, im Falle eines Angriffs den Schiedskontext in das Protokoll einzubinden.

Unter der gleichzeitigen Verwendung des prüfsummenbasierten Protokolls mit Schiedskontext und eines verschachtelten Routekonzepts ist die einzig sinnvolle Koalition aus der Menge der möglichen Koalitionen zweier böswilliger Kontexte diejenige des aktuellen Arbeitskontexts mit seinem Nachfolgerkontext. Der Nachfolgerkontext wird seine Mittäterschaft allerdings verweigern, da er sonst von seinem eigenen Nachfolger der Bösartigkeit überführt würde.

Unter Nutzung des prüfsummenbasierten Nichtabstreitbarkeits-Protokolls mit Schiedskontext veranlassen obige Überlegungen den Nachfolgerkontext ebenfalls, von einer Mittäterschaft abzusehen. Zur Brechung dieses Protokolls wäre ein weiterer sinnvoller und möglicher Koalitionspartner des aktuellen Arbeitskontexts sein Vorgängerkontext. Der Vorgängerkontext verfügt in Form des temporären Schlüssels über protokollspezifisches Wissen. Würde er diesen Schlüssel jedoch preisgeben, ohne gleichzeitig vom aktuellen Arbeitskontext eine Unschuldsquittung zu erhalten, dann würde der Schiedskontext ihn als den Täter überführen. Koalierende Angriffe dieser beiden böswilligen Partner brechen das Protokoll daher nur, wenn zumindest der Vorgängerkontext dem aktuellen Arbeitskontext vertraut. Existiert ein Vertrauensverhältnis der beiden böswilligen Koalitionspartner nicht zumindest in dieser Richtung, so kann auch eine derartige Koalition das Protokoll nicht brechen.

6.2 Ausblick

Um einen umfassenden Schutz von Arbeitskontext und Arbeitsrechner zu gewährleisten, sind Schutzmechanismen zur Überwachung der Kommunikation von Agenten unverzichtbar. Andernfalls könnten Koalitionen böswilliger Agenten $\dot{A}, \ddot{A}_1, \dots, \ddot{A}_m \rightarrow k$ die Zugriffs-/Zugangskontrolle des Arbeitsrechners überlisten. Ein Agent, der mit