Sicherheitsaspekte in Active Directory

Vortrag für den Workshop
„Sicherheit in vernetzten Systemen“
am 4. und 5. März 1998 in Hamburg

Dr. Ralf Schneider, DFN–Projekt AMBIX,
Zentrum für Datenverarbeitung, Universität Tübingen
ambix-d@mail500.uni-tuebingen.de

Active Directory

Active Directory (AD) [1, 2, 3] wird als zentraler Bestandteil in Microsofts Betriebssystem Windows NT 5.0 enthalten sein.

Mit AD wird nicht nur eine Anbindung an andere Directory–Dienste ermöglicht, vielmehr soll damit auch die gesamte Ressourcen– und Nutzerverwaltung des Betriebssystems realisiert werden. Durch die Vereinigung aller bisher bekannter Windows basierter Verzeichnis–Dienste und die Hinzunahme neuer Features soll AD sowohl als Informationsquelle als auch als Administrations–Tool dienen.

1 Einführung in Active Directory

AD baut auf den Datenmodell des X.500 [4 – 12] auf. Für ein besseres Verständnis sollen hier zunächst einige Grundbegriffe des X.500 vorgestellt werden.

1.1 Grundbegriffe des X.500

Das X.500 stellt eine von internationalen Normierungsgremien (ITU und ISO) standardisierte weltweit verteilbare und weltweit zugängliche Datenbank zur Verfügung. Im X.500 können beliebige, hierarchisch darstellbare Informationen gespeichert werden.

Alle Daten werden in Servern, den *Directory System Agents* (DSAs) vorgehalten und verwaltet. Mittels *Directory User Agents* (DUAs) kann auf die Daten zugegriffen werden, um diese Daten zu suchen, anzuzeigen, zu verändern, zu löschen oder neue anzulegen. Speziell für die Kommunikation zwischen DSA und DSA sowie DSA und DUA sind im X.500–Standard verschiedene Protokolle definiert worden. Diese sind

- *Directory Access Protocol* (DAP) mit der DUA–Anfragen an einen DSA weitergeleitet werden.
- *Directory System Protocol* (DSP) zur DSA ↔ DSA Kommunikation, z.B. Weiterleitung von DUA–Anfragen zwischen kooperierenden DSAs, die vom befragten DSA nicht direkt beantwortet werden können.
- *Directory Operational Binding Management Protocol* (DOP) zum Austausch der für eine Kooperation zwischen DSAs benötigten Informationen, z.B. für die Replikation.
- *Directory Information Shadowing Protocol* (DISP) zum Austausch der zur Replikation vorgesehenen Informationen.

Zusätzlich zu den im X.500 definierten Protokollen hat sich inzwischen das *Lightweight Directory Access Protocol* (LDAP) [13] der University of Michigan als eines der wichtigsten DUA/DSA Protokolle etabliert.

Die von den DSAs gespeicherten Einträge sind hierarchisch strukturiert im *Directory Information Tree* (DIT) angeordnet, wobei jeweils ein sog. Master–DSA für die Verwaltung eines definierten Teilbereiches des DITs zuständig ist.

Jeder Eintrag im Directory, ein *Object*, setzt sich aus einem oder mehreren Attributen zusammen, wobei jedes dieser Attribute einen oder mehrere Attribut–Werte besitzen kann, die jeweils einer genau definierten Attribut–Syntax unterliegen. Ein spezielles Attribut ist die *ObjectClass*, diese spezifiziert den Typ eines Eintrags näher und legt fest, welche Attribute zwingend erforderlich und welche optional vorhanden sein können.

Bestimmte Attribut–Werte bilden den innerhalb einer Hierarchieebene eindeutigen *Relative Distinguished Name* (RDN) eines Eintrags. Zusammen mit den im DIT hierarchisch übergeordneten RDNs ergibt sich daraus der im gesamten DIT eindeutige *Distinguished Name* (DN) dieses Eintrags (vgl. Abb. 1).

Als eine Hauptanwendung des X.500 ist die Speicherung von Organisations– und Personendaten zu nennen. Alle hierfür benötigten Objektklassen (z.B. *Organization* (O), *Organizational Unit* (OU)) und Attribute (z.B. *Common Name* (CN)) werden vom Standard zur Verfügung

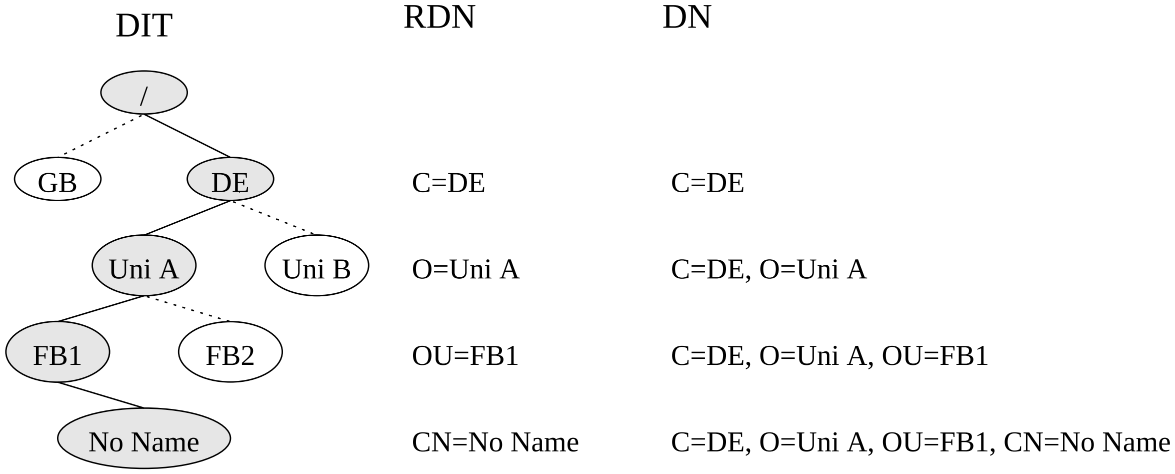

Abbildung 1: *X.500 Namen, Zusammenhang zwischen DIT, RDN und DN*

gestellt. Die zur Verfügung stehenden Objektklassen und ihre Attribute werden im *Schema* definiert. Um lokalen Anforderungen gerecht zu werden, kann das *Schema* durch die Definitionen von neuen Objektklassen und Attributen erweitert werden.

Mit *Naming Context* wird ein zusammenhängender Teilbaum des DITs, zusammen mit der Angabe des Eingangs–DN dieses Baumes und aller seiner Referenzen zu hierarchisch über– bzw. untergeordneten Einträgen bezeichnet. Z.B. werden Bereiche des DITs die auf andere DSAs verteilt bzw. repliziert werden sollen durch einen *Naming Context* definiert.

Darüberhinaus werden im X.500 Mechanismen zur Zugriffskontrolle und Authentifizierung definiert, die bereits in einem Beitrag zum 4. Workshop des Projekts „CERT im DFN“ vorgestellt wurden (vgl. [14]).

1.2 Grundbegriffe in Active Directory

Neben Begriffen aus dem Sprachgebrauch des X.500 werden in AD weitere Begriffe definiert, die hier kurz angeführt werden sollen.

Ein Objekt welches andere Objekte enthalten kann wird als *Container* bezeichnet, *Virtual Containers* ermöglichen den Zugang zu anderen Directory–Diensten. Die Hierarchie von Objekten wird im *Tree* (vgl. DIT im X.500) wiedergegeben, Knoten eines *Trees* sind *Container*.

Mehreren Domänen, die *Schema* und Konfiguration teilen sowie einen kontinuierlichen Namensraum bilden, werden als *Domain Tree* bezeichnet. Die Domänen in einem *Domain Tree* sind durch Sicherheitsbeziehungen miteinander verbunden. Im Gegensatz dazu bezeichnet der *Forest* mehrere solcher *Domain Trees* die keinen gemeinsamen Namensraum bilden, aber *Schema* und Konfiguration teilen.

Eine *Site* bezeichnet den Ort innerhalb eines Netzwerks an dem AD–Server lokalisiert sind, diese kann mehrere TCP/IP Subnetze umfassen.

2 Features von Active Directory

2.1 Zugang zum Active Directory

Der Zugang zum AD kann mit verschiedenen Protokollen realisiert werden.

LDAP (v2 und v3) wird von AD als zentrales Protokoll unterstützt und soll die Interoperabilität über verschiedene Plattformen und Betriebssysteme hinweg gewährleisten.

Weiterhin ist ein Zugang über eine *Remote Procedure Call* (RPC) Schnittstelle möglich. Obwohl das AD auf dem X.500 Modell basiert, sollen die X.500 Protokolle DAP, DSP, DISP und DOP vorerst nicht implementiert werden. Bei Bedarf soll jedoch ein Subset zur Verfügung gestellt werden.

Mit den folgenden drei *Application Programming Interfaces* (APIs) stehen einfache Programmierschnittstellen zu AD bereit: *Active Directory Service Interface* (ADSI) [15], LDAP C API [16] und *Windows Messaging* API (MAPI).

Das ADSI stellt ein objektorientiertes Interface zur Verfügung, welches auch für verschiedene andere Directory Dienste (Windows NT 4.x, Novell NetWare 3.x und 4.x sowie alle mit LDAP Unterstützung) angeboten wird und mit verschiedenen Programmiersprachen (z.B. Java, VisualBasic, C/C++) angesprochen werden kann.

Mit ADSI sollen die speziellen Fähigkeiten der Directory Dienste verschiedener Anbieter abstrahiert werden und somit die Entwicklung und Verwaltung verteilter heterogener Systeme erleichtern.

Vorhandene LDAP–Applikationen, die auf dem LDAP C API basieren, sollten ohne bzw. mit nur geringen Änderungen, die sich lediglich auf die Erweiterung AD spezifischen Objektklassen und Attribute beschränken, mit AD zusammenarbeiten. Mit der MAPI–Unterstützung, soll bereits existierenden MAPI–Applikationen die Zusammenarbeit mit AD ermöglicht werden. Bei der Entwicklung neuer AD–Anwendungen sollte jedoch auf ADSI bzw. LDAP C API zurückgegriffen werden.

2.2 Neuer Namensraum

In der Konzeption von AD bildet jede Windows NT Domain einen *Tree*, dieser kann so in OUs aufgeteilt werden, daß die hierarchische oder logische Struktur der betreffenden Organisation wiedergegeben wird. Die einzelnen Domains können schließlich zu einem *Domain Tree* zusammengefaßt werden (vgl. Abb. 2).

Mit AD wird in Windows NT 5.0 ein hierarchisch strukturierter Namensraum zur Verfügung gestellt, der das flache Domain–Konzept früherer Windows NT Versionen durch einen beliebig skalierbaren *Domain Tree* ersetzt.

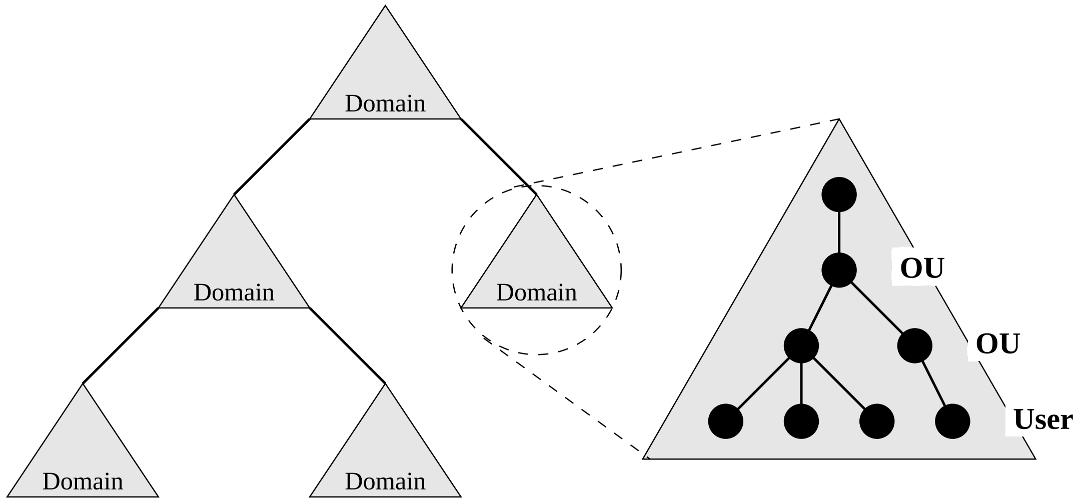

Abbildung 2: *Domain Tree und OUs, Struktur eines Active Directory*

2.3 Replikation

Im allgemeinen treten bei Directory Diensten Lesezugriffe wesentlich häufiger auf als Schreibzugriffe, typisch etwa 100:1. Die Replikation der Daten auf mehrere Server kann daher zu einer deutlich gesteigerten Performance bei Suchanfragen führen.

Die Verteilung der Daten erfolgt bei AD mit der sog. *Multi Master Replication*. Im Gegensatz zu Windows NT 4.0, bei der zwischen *Primary Domain Controler* (PDC), dieser besitzt Schreib– und Leserechte, und *Backup Domain Controler* (BDC), die nur Leserechte besitzen, unterschieden wurde, existieren bei AD nur *Domain Controler* (DC), die alle gleichgestellt sind. Objekte können auf jedem beliebigen DC erzeugt, verändert oder gelöscht werden.

Für die Replikation in AD werden anstelle von Zeitmarken *Update Sequence Numbers* (USNs; 64–Bit Zahl) verwendet. Bei jeder Änderung eines Attributs wird automatisch die entsprechende USN erhöht und zusammen mit dem geänderten Attribut abgespeichert. Jeder DC hält eine Tabelle der eigenen USNs und der bisher von seinen Replikationspartnern erhaltenen USNs vor. Bei der Replikation werden nun von jedem anderen DC alle Änderungen angefordert, deren USN größer ist als die jeweils zuletzt erhaltene USN dieses DCs.

Wird ein Attribut erzeugt und zum ersten mal ins AD geschrieben, wird auch eine Versionsnummer für dieses Attribut erzeugt. Bei Änderungen des Attributs wird diese Nummer nur auf dem System erhöht, welches die Änderung initiiert hat. Die Replikation hat keinen Einfluß auf die Versionsnummer. Zusätzlich wird bei jeder Änderung eines Attributs ein Zeitstempel erzeugt und mit abgespeichert.

Werden Attribute auf mehreren DCs verändert, bevor deren Replikation vollständig abgeschlossen ist, kommt es zu Kollisionen, die über die Versionsnummer der jeweiligen Attribute aufgelöst werden können. Es wird die jeweils höchste Versionsnummer verwendet. Bei identischen Versionsnummern wird dasjenige Attribut verwendet, welches den späteren Zeitstempel besitzt. Sind auch diese identisch, wird ein Vergleich der Buffergröße durchgeführt, wobei das Attribut mit dem größeren Buffer gewinnt. Der Zeitstempel fungiert hier nur als Entscheidungskriterium, um eines der beide Attribute auszuwählen. Eine genaue Zeitsynchronisation zwischen den einzelnen Rechnern wird daher nicht benötigt.

Da die Wiedereingliederung geänderter Einträge auf Attribut– und nicht auf Objektebene erfolgt, sollte die Anzahl solcher Kollisionen auf ein Minimum reduziert sein. Inwieweit diese Voraussage zutrifft, muß sich in der Praxis zeigen.

Alle Vorgänge werden protokolliert, so daß ein Administrator jederzeit die Möglichkeit hat, einen früheren Zustand wiederherzustellen und die Replikation mit zuvor abgewiesenen Daten erneut durchzuführen.

2.4 DNS–Integration und Namen

In AD werden Teile des *Domain Name Service* (DNS) und des X.500 miteinander kombiniert. Die einzelnen Server veröffentlichen ihre Adresse im DNS mittels Dynamic–DNS [17]. Die dort in *Service Resource Records* (SRV RRs) gespeicherten Daten ermöglichen die Zuordnung des Namens eines Dienstes zu der Adresse des anbietenden Rechners. Die allg. Form eines solchen SRV RRs wird wiedergegeben durch:

<service>.<protocol>.<domain>

also z.B. den LDAP Service über TCP

ldap.tcp.<domain>

Neben der Möglichkeit Einträge mittels ihres DNs zu benennen, wird in AD auch die Suche nach Attribut–Werten unterstützt. Um dabei die Suche zu erleichtern, sind zwei spezielle Attribute definiert worden; der *Object Globally Unique Identifier* (GUID) sowie der *User Principal Name* (UPN).

Jedem Objekt wird bei seiner Erzeugung eine eindeutige GUID (128–Bit Zahl) zugewiesen, diese GUID wird niemals verändert. D.h. Einträge können auch noch über die GUID gefunden werden, nachdem sich ihr DN geändert hat.

Der UPN ist ein *frindly name* nach RFC 822 [18], der sich aus dem Benutzernamen und der Domain, auf der Benutzer angesiedelt ist, zusammensetzt. So hat z.B. Karl Mustermann in der Domain zdv.uni-tuebingen.de den UPN `KarlMustermann@zdv.uni-tuebingen.de`.

Weiterhin werden HTTP URLs der Form `HTTP://somedomain/path-to-page`, z.B.

`HTTP://server.uni-tuebingen.de/zdv/KarlMustermann`

LDAP URLs und X.500 Namen, z.B.

```
LDAP://ldap.uni-tuebingen.de/cn=Karl%20Mustermann,ou=zdv,
o=Universitaet%20Tuebingen,c=DE
```

sowie Namen nach der *Universal Naming Convention* (UNC) unterstützt, z.B.

`\\server.uni-tuebingen.de\Printer\PS\ps001`

UNC–Namen werden von Windows NT für gemeinsam benutzte Ressourcen wie z.B. Drucker und Dateien benutzt.

2.5 Global Catalog

Im allgemeinen wird das AD auf mehrere Server verteilt sein. Der DN eines Eintrags enthält jedoch alle Informationen die zu seiner Lokalisierung benötigt werden.

Der *Global Catalog* (GC) ermöglicht auch in verteilten ADs das schnelle Auffinden von Einträgen, bei denen einzelne Attribut–Werte, nicht aber der DN bekannt sind. Im GC werden alle Objekte im AD repliziert, jedoch jeweils nur mit einem kleinen Subset ihrer Attribute. Dies sind die Attribute, die am häufigsten gesucht werden (Vor– und Nachname, Benutzername usw.).

Dieser GC wird automatisch vom AD Replikationssystem erzeugt. Das für die Erzeugung des GC verwendete Basis–Set von Attributen kann jederzeit erweitert werden.

3 Sicherheit im Active Directory

Die gesamte Nutzer– und Ressourenverwaltung sowie die Speicherung aller sicherheitsrelevanten Informationen, wie domainweit gültige Paßwörter und Zugangsrechte sowie Informationen über die Sicherheitsbeziehungen zu anderen Domänen oder *Forests*, soll mit AD realisiert werden.

Vom Betriebssystem wird ein eindeutiges und einheitliches Sicherheitsmodell für alle Einträge im AD implementiert, welches auf der Zugehörigkeit zu einer bestimmten Gruppe basiert. Bei der Authentifizeirung angebenene Informationen werden mit den Einträgen im AD verglichen, um dann die entsprechenden Zugriffsrechte zu gewähren. Daher besteht bei Windows NT 5.0 eine fundamentale Beziehung zwischen AD und Sicherheit, die nur durch die vollständige Integration von AD und Betriebssystem realisiert werden konnte (vgl. hierzu auch [1, 2, 3] und [19]).

3.1 Sicherheitsbeziehungen zwischen Domains

Die Form des in Abschnitt 2.2 angesprochenen *Domain Trees* und die Beziehungen der einzelnen Domänen zueinander werden durch die *Domain Trust Relationship* bestimmt. Diese Vertrauensbeziehungen erlauben die Authentifizierung von Benutzern einer Domain

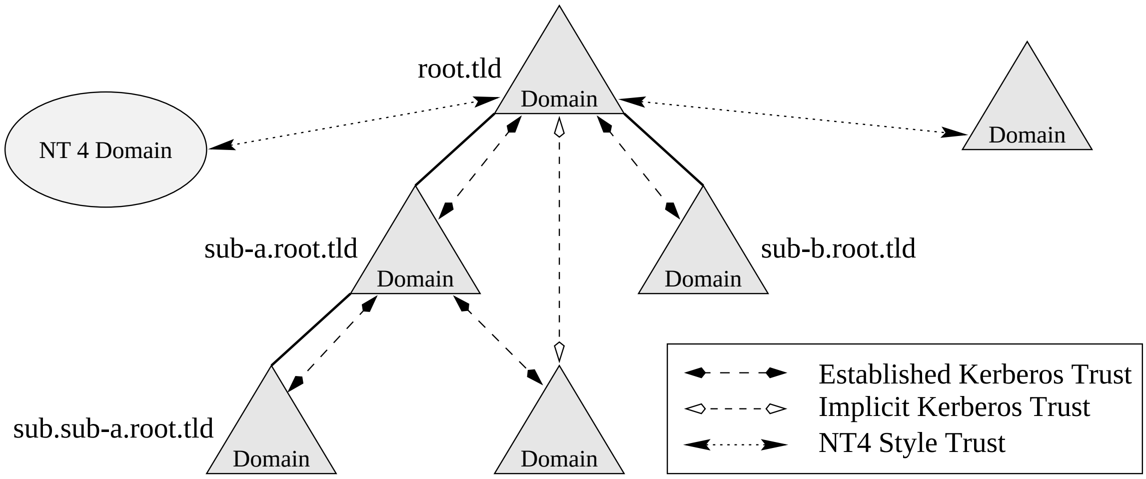

Abbildung 3: *Sichtweisen und Sicherheitsbeziehungen in Active Directory*

durch den AD–Server einer anderen vertrauenswürdigen Domain im selben *Domain Tree* oder *Forest*.

Windows NT 5.0 unterstützt zwei Arten dieser Beziehungen, eine explizite *one–way* Beziehung (NT4–style) und eine transitive auf dem Kerberos–Protokoll [20] basierte.

Zwischen den Domänen desselben *Domain Trees* wird automatisch der Kerberos Trust etabliert (vgl. Abb. 3). Dies vereinfacht zwar das Management der Sicherheitsbeziehungen zwischen den einzelnen Domänen eines Baumes, birgt aber auch gewisse Sicherheitsrisiken. Gibt es innerhalb eines Baumes eine weniger vertrauenswürdige Domäne, kann alternativ der NT4–style Trust definiert werden.

Die Hierarchie der Sicherheistbeziehungen zwischen den einzelnen Domänen wird in speziellen Objekten im AD abgespeichert. Bei Beziehungen zu Domänen eines anderen Baumes (im *Forest* oder außerhalb) wird zusätzlich der DN der entsprechenden Domäne vorgehalten. In diesem Fall ermöglicht ein spezielles Flag die Unterscheidung zwischen Domänen im selben *Forest* und solchen die außerhalb des *Forests* angesiedelt sind.

3.2 Administrations– und Zugriffsrechte

Sicherheitsrelevante Informationen müssen vor unerlaubter Veränderung geschützt werden. Im objektbasierten Sicherheitsmodell von AD werden alle Zugriffsrechte auf die Attribute eines Eintrags mit *Access Controll Lists* (ACLs) geregelt. Zur Authentifizierung und damit der Entscheidung ob geforderte Zugriffsrechte gewährt werden können, wird auf die Zugriffskontrollmechanismen von Windows NT zurückgegriffen. D.h. die Authentifizierung einer Client– oder Benutzeranfrage erfolgt auf Betriebssystemebene.

Zugriffsrechte können mit den ACLs auf verschiedenen Ebenen gewährt bzw. verboten werden. So kann z.B. bestimmt werden, ob vergebene Rechte wie Suchen, Anzeigen aber auch Ändern, Löschen oder Neuanlegen für einen gesamten Eintrag oder *Container*, eine frei definierbare Gruppe von Attributen innerhalb eines Eintrags oder nur für einzelne Attribute des Eintrags Gültigkeit besitzen sollen.

Sofern für einen Eintrag bzw. *Container* keine eigene ACL definiert wird, erbt dieser bei seiner Erzeugung die ACL des im AD hierarchisch übergeordneten Eintrags. Änderungen an der ACL eines Eintrags werden nicht automatisch an die im AD hierarchisch untergeordneten Einträge vererbt. Geänderte ACLs müssen explizit im Baum propagiert werden. Die Verifikation der Zugriffsrechte ist eine wichtige und häufig auftretende Operation im AD. Mit diesem statischen Modell der Vererbung kann diese Operation optimiert und dadurch sehr schnell durchgeführt werden.

Als eines der wichtigsten Sicherheitsfeatures von AD bezeichnet Microsoft die sog. *Delegation*. Dahinter verbirgt sich letztlich nur die Möglichkeit mit Hilfe der ACLs einem einzelnen Benutzer die Administration eines bestimmten Teilbereichs des ADs oder definierbarer Aufgaben zu übertragen. Ein globaler Domain–Administrator mit umfassenden Systemprivilegien wird überflüssig.

Rechte können z.B. so vergeben werden, daß ein lokalen Administrator das Passwort eines Benutzer verändern kann, nicht aber neue Benutzer anlegen oder Benutzer löschen.

4 Authentifizierung

Da sich die Sicherheitsanforderungen verschiedener Organisationen stark voneinander unterscheiden, werden bei Windows NT 5.0 verschiedene Verfahren zur Benutzerauthentifizierung zur Verfügung gestellt.

Das in früheren Versionen von Windows NT ($\leq$ Windows NT 4.0) verwendete Windows NT *Lan Manager* (NTLM) Protokoll wird weiterhin unterstützt, um *remote file access* und eine authentifizierte RPC Verbindung auch mit diesen älteren Versionen von Windows NT zu ermöglichen.

Die Version 5 des Kerberos Protokoll wird NTLM als primäres Authentifizierungsprotokoll ersetzen. Kerberos ist ein ausgereifter Standard, einer der Vorteile ist die gegenseitige Authentifizierung beider Partner (Client und Server), was zu einer Reduzierung der Serverlast während des Verbindungsaufbaus führt.

Beim Kerberos Protokoll wird ein sogenannter Key verwendet, der ein Geheimnis darstellt, das nur dem Kerberos–Server und dem jeweiligen Anforderer bekannt ist. Für Windows NT 5.0 will Microsoft dieses Protokoll so erweitern, daß auch eine Authentifizierung mit auf asymmetrischer Verschlüsselung basierenden Verfahren durchgeführt werden kann. Ein diesbezüglicher Erweiterungsvorschlag wurde der entsprechenden IETF–Arbeitsgruppe CAT *(Common Authentication Technology)* vorgelegt.

Distributed Password Authentication (DPA) ist ein Sicherheitsprotokoll, das von verschiedenen Internet–Providern (z.B. MSN und CompuServ) verwendet wird. DPA ermöglicht Be-

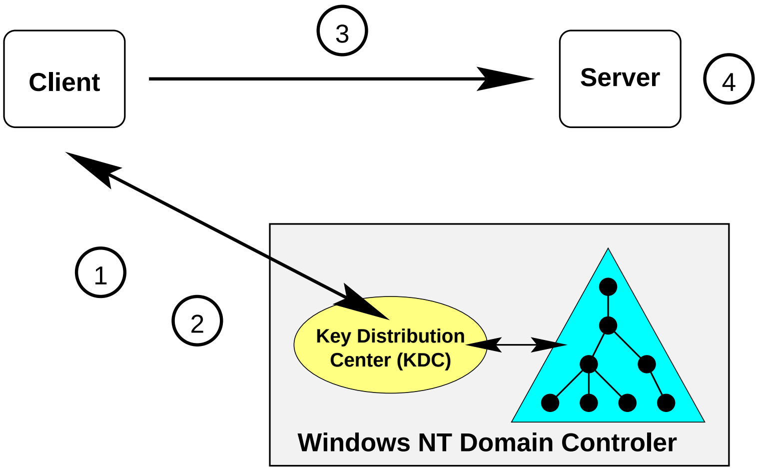

Abbildung 4: *Realisierung der Kerberos Authentifizierung bei Windows NT 5.0*

nutzern die Verwendung desselben Passworts für alle Rechner innerhalb der Domain des entsprechenden Providers.

Nicht zuletzt werden auf asymmetrischen Verschlüsselungsverfahren (PKCS = *Public Key Crypto System*) beruhende Methoden unterstützt. Zu nennen sind hier vorallem *Secure Socket Layer* (SSL) und X.509v3. Bestimmte Zertifikatinformationen werden mit den Einträgen im AD verglichen und so eine Authentifizierung des Clients (SSL) oder externen Benutzers (X.509v3) durchgeführt. Aus Sicherheitsgründen erfordert diese Form der Authentifizierung die aktive Konfiguration von *Certification Authorities* (CAs) deren Zertifikate als vertrauenswürdig akzeptiert werden.

Die PKCS erfordern das Management der Zertifikate sowie der Zertifikat–Rückruf–Listen (*Certificate Revocation Lists* = CRLs), dafür soll in Windows NT 5.0 der Microsoft Certificate Server enthalten sein.

4.1 Kerberos Authentifizierung und NT

Bei Windows NT 5.0 wird auf jedem DC auch ein *Key Distribution Center* (KDC) implementiert. Das AD wird dabei als Datenbank zum Abruf und zur Speicherung aller für die Authentifizierung relevanten Daten verwendet.

Wenn sich ein Benutzer bei seiner NT–Workstation anmeldet, erhält er vom KDC ein erstes Ticket für die aktuelle Sitzung (vgl. Abb. 4 (1)), welches im lokalen Ticket–Cache des Rechners gespeichert wird.

Fordert nun ein Benutzer oder Client einen Service von einem Server an, wird zuerst geprüft, ob für diesen Server ein gültiges Ticket im lokalen Cache vorliegt. Ist dies nicht der Fall, so wird vom KDC ein neues Sitzungs–Ticket für diesen Server angefordert (vgl. Abb. 4 (2)). Dieses neue Ticket wird dann ebenfalls im lokalen Cache für eine evtl. spätere Verwendung gespeichert. Die Gültigkeitsdauer diese Tickets kann, abhängig von der geltenden Sicherheits–Politik, individuell festgelegt werden.

Verliert ein Sitzungs–Tickets seine Gültigkeit, bevor die akutelle Sitzung beendet wurde, wird eine entsprechende Fehlermeldung zurückgegeben. Abhängig von geltenden Sicherheits–Politik kann dann automatisch ein neue neues Ticket angefordert und die Sitzung fortgeführt oder diese (Sitzung) auch abgebrochen werden.

Nachdem der Client das benötigte Sitzungs–Ticket erhalten hat, wird es dem entsprechenden Server übermittelt (vgl. Abb. 4 (3)). Das Ticket enthält neben anderen Informationen (z.B. Zeit, Datum) auch den geheimen Server–Schlüssel und kann daher vom Server, ohne daß dieser eine Verbindung zum KDC herstellen muß, auf seine Gültigkeit hin überprüft werden (vgl. Abb. 4 (4)).

Active Directory und AMBIX

5 AMBIX

Das DFN–Projekt AMBIX[1] ([A]ufnahme von [M]ail–[B]enutzer [i]n das [X].500 Directory) ermöglicht Organisationen aus dem Bereich der Forschung in Deutschland, bequem und datenschutzkonform kommunikationsrelevante Personendaten im X.500 zu veröffentlichen und aktuell zu halten, ohne selbst einen DSA betreiben zu müssen.

Auf Basis einer vom DFN–Verein in Auftrag gegebenen Datenschutzexpertise wird von den beteiligten Organisationen ein Minimalset an personenbezogenen Daten, bestehend aus Vor– und Nachname, Organisationszugehörigkeit, akademischer Titel, dienstlicher Telefon– und Faxnummer sowie der E–Mail–Adresse, an das Projekt geliefert. Die betroffenen Personen werden über die Lieferung Ihrer Daten und Ihre Rechte (Löschung, Sperrung und Änderung der Daten) informiert und nach einer Widerspruchsfrist von 6 Wochen im AMBIX–Verzeichnis veröffentlicht, sofern Sie der Veröffentlichung nicht widersprochen haben.

Für Angehörige von Organisationen, die sich nicht aktiv am Projekt AMBIX beteiligen, wurde die Möglichkeit des Selbsteintrags ins X.500 geschaffen.

Aufgrund der beschriebenen Vorgehensweise dürfen Personendaten nur an Länder weitergegeben werden, in denen der deutschen Datenschutzgesetzgebung vergleichbare Bestimmungen gelten. Alle erforderlichen Mechanismen zur Realisierung der vorgegebenen Zugangsbeschränkungen sind in TWEB (Tübinger Web–Gateway), einem WWW–X.500–Gateway, implementiert. Personendaten werden nur an TWEB weitergegeben. Die Ermittlung des Landes aus dem eine Abfrage erfolgt wird an Hand der *Top Level Domain* (TLD) des anfragenden Rechners durchgeführt. Dafür muß der abfragende Rechner im DNS eingetragen sein und

[1] gefördert aus Mitteln des BMBF

seine TLD der eines datenschutztreibenden Lands entsprechen. Rechner mit TLDs die keinen Rückschluß auf ein Land zulassen (wie z.B. .com), werden vom Zugriff auf Personendaten ausgeschlossen.

Ein Zugang über die in neueren Browserversionen implementierte LDAP–Schnittstelle ist noch nicht möglich, da diese Clients bisher nur einen *anonymous bind*, d.h. ohne Angabe von DN und Passwort, beim LDAP–Server vornehmen, so daß die vorgegebenen Zugriffsbeschränkungen nicht durchgesetzt werden können.

In Zusammenarbeit mit dem DFN–PCA–Projekt an der Universität Hamburg wird eine sichere Kommunikationsstruktur mit den von der PCA zertifizierten CAs aufgebaut. Die von den CAs ausgestellten Benutzer–Zertifikate sowie die CRLs sollen an AMBIX übermittelt und im X.500 veröffentlicht werden (vgl. [14]).

6 Integration von AMBIX in Active Directory

Eine Integration von AMBIX in AD kann auf verschiedenen Ebenen erfolgen.

Der Zugriff von AD auf AMBIX über die LDAP–Schnittstelle ist die unterste Ebene, auf der eine solche Integration erfolgen kann. Hierbei sind jedoch die durch die Datenschutzexpertise auferlegten Beschränkungen bei der Weitergabe von Personendaten zu berücksichtigen. Die Zugriffsbeschränkungen können z.B. auf Protokollebene mit den von LDAP zur Verfügung gestellten Authentifizierungsverfahren realisiert werden, würden dann aber nur eingetragenen Benutzern einen Zugang zum AMBIX–Verzeichnis ermöglichen.

Zur Bereitstellung der von den CAs ausgestellten Zertifikaten und CRLs wird von AMBIX ein X.500–Zertifikat–Verzeichnisdienst aufgebaut. Alternativ kann dieser Verzeichnisdienst aber auch mit AD realisiert werden. Der Zugriff auf die Zertifikate aus dem X.500 bzw. AMBIX–Verzeichnis würde dann über eine LDAP–Schnittstelle und einen *Virtual Container* im AD realisiert.

Schließlich kann auch die Möglichkeit in Betracht gezogen werden, den gesamten AMBIX–Dienst (Personen–, Organisations– sowie Zertifikat–Verzeichnis) vollständig in AD zu integrieren. Von AMBIX verwaltete Organisationen werden direkt in mehreren AD–Servern vorgehalten. Auf Organisationen, die einen eigenen DSA pflegen, wird vom AMBIX–Verzeichnis aus verwiesen. Da von AD bisher keine „echten“ X.500 Protokolle unterstützt werden, müßten entsprechende Verweise über einen *Virtual Container* und LDAP implementiert werden.

Inwieweit sich diese Möglichkeiten der Integration von AMBIX und AD praktisch realisieren lassen, bleibt zu prüfen. Zum Zeitpunkt der Entstehung dieses Artikels standen von Windows NT 5.0 und AD leider nur Beta–Realeases mit eingeschränkter Funktionalität zur Verfügung, weswegen noch keine praktischen Erfahrungen gesammelt werden konnten.

Literaturverzeichnis

[1] *Active Directory Technical Summary*, White Paper, Microsoft Corporation, 1997.

[2] *Microsoft Windows NT Active Directory: An Introduction to the Next Generation Directory Services*, White Paper, Microsoft Corporation, 1997.

[3] *Directory Integration with the Windows NT 5.0 Active Directory*, White Paper, Microsoft Corporation, 1997.

[4] X.500: *The Directory: Overview of Concepts, Models, and Services*, (ISO/ICE 9594–1:1993)

[5] X.501: *The Directory: Models*, (ISO/IEC 9594–2:1993).

[6] X.511: *The Directory: Abstract Service Definition*, (ISO/IEC 9594–3:1993).

[7] X.518: *The Directory: Procedures for Distributed Operation*, (ISO/IEC 9594-4:1993).

[8] X.519: *The Directory: Protocol Specifications*, (ISO/IEC 9594–5:1993).

[9] X.520: *The Directory: Selected Attribute Types*, (ISO/IEC 9594–6:1993).

[10] X.521: *The Directory: Selected Object Classes*, (ISO/IEC 9594–7:1993).

[11] X.509: *The Directory: Authentication Framework*, (ISO/IEC 9594–8:1993).

[12] X.525: *The Directory: Replication*, (ISO/IEC 9594–9:1993).

[13] W. Yeong, T. Howes and S. Kille, RFC 1777: *Lightweight Directory Access Protocol*, März 1995.

[14] K.P. Gietz, *Sicherheitsaspekte im X.500 und im Projekt AMBIX*, Workshop des Projekts „CERT im DFN“, 4. und 5. März 1997, Hamburg, DFN–Bericht Nr. 84, DFN–CERT (Hrsg.), Juli 1997, S. C1 ff.

[15] *Microsoft Active Directory Service Interface: ADSI*, White Paper, Microsoft Corporation, 1997.

[16] T. Howes and M. Smith, RFC: 1823: *The LDAP Application Program Interface*, August 1995.

[17] P. Vixie, S. Thomson, Y. Rekhter and J. Bound, RFC 2136: *Dynamic Updates in the Domain Name System (DNS UPDATE)*, April 1997.

[18] David H. Crocker, RFC 822: *Standard for the Format of ARPA Internet Text Messages*, August 1982.

[19] *Secure Networking Using Microsoft Windows NT 5.0 Distributed Security Services*, White Paper, Microsoft Corporation, 1997.

[20] J. Kohl und C. Neumann, RFC 1510: *The Kerberos Network Authentication Service (V5)*, September 1993.