# Vergabe von *root*-Rechten an UNIX-Benutzer besser verwalten

Arturo Lopez

Hewlett Packard GmbH
Arturo.Lopez-Ayala@hp.com

## 1 Einleitung

Das Sicherheitsmodell von UNIX für die Delegation von*root*-Rechten basiert auf dem Prinzip „Alles oder nichts“. Dieses Modell bringt in der heutigen Zeit, in der die UNIX-Rechner vernetzt sind, ein großes Problem mit sich, denn wer das *root*-Passwort kennt, hat die komplette Kontrolle über den Rechner. Eine Unterscheidung zwischen unterschiedlichen Rollen oder Identitäten ist nicht möglich. Denn ein Help Desk-Mitarbeiter, der lediglich ein paar wenige Befehle mit *root*-Rechten für seine tägliche Arbeit braucht, bekommt bei diesem Modell die gleichen Rechte wie ein Systemverwalter, der die komplette Kontrolle über das System für seine Arbeit benötigt, nämlich alle. Ein Help Desk-Mitarbeiter hat somit Zugang zu vertraulichen Daten und kann absichtlich oder unabsichtlich das System verändern.

Eine Zuordnung der ausgeführten Befehle und Programme zu den jeweiligen Mitarbeitern oder Rollen ist nicht möglich, denn alle Befehle werden vom *root account* ausgeführt. Diese Systeme sind kaum administrierbar und würden eine Revision nicht bestehen. Eine einfache Protokollierung kann man erzielen, indem die Mitarbeiter, die *root*-Rechte brauchen, zur Gruppe *system* hinzugefügt werden. Eine Protokollierung der einzelnen, ausgeführten Befehle ist jedoch nicht möglich. Ferner hat ein Benutzer, der zur Gruppe *system* gehört, alle Rechte des *root accounts*. Die Sicherheitslücke bleibt also bestehen.

Das Sicherheitsmodell von UNIX für die Vergabe von *root*-Rechten an UNIX-Benutzer sollte, um den heutigen Anforderungen gerecht zu werden, um folgendes erweitert werden:

- granulare Vergabe von *root*-Rechten für die Ausführung von Programmen
- Protokollierung der mit *root*-Rechten ausgeführten Befehle und des Benutzers, der den Befehl ausgeführt hat
- Unterstützung von Rollen, z. B. Help Desk-Mitarbeiter, oder Operatoren
- zentrale Konfiguration und zentrale Verwaltung der Sicherheitsrichtlinien
- Ausfallsicherheit
- Unterstützung von Systemen, die unternehmensweit eindeutige Identitäten gewährleisten (Identity Management Systeme).

In diesem Artikel stellen wir Super User Do (SUDO), Powerbroker von der Firma Symark und OpenView Select Access sowie OpenView Select Identity (Identity Management-Produkte von Hewlett Packard) vor, diese Produkte stellen in unterschiedlichen Ausprägungen eine Lösung für das Dilemma zur Delegation von *root*-Rechten auf UNIX-Systeme dar.

## 2 Super User Do (SUDO)

SUDO ist eine Free Software, welche nach dem ISC-Lizenzierungsmodell angeboten wird. Mit SUDO ist es möglich, einem Benutzer die Berechtigung zu erteilen, bestimmte Programme mit *root*-Rechten auszuführen. Die Berechtigung kann aus einer Kombination von Rechner-, Benutzer- und Befehlsnamen zusammengestellt werden. Der Benutzer authentifiziert sich mit seinem Betriebssystempasswort SUDO gegenüber; nach erfolgreicher Authentifizierung wird ein Ticket für die Gültigkeitsdauer der Authentifizierung (in der Regel 5 Minuten) ausgestellt. Der Befehl kann ausgeführt werden, falls er genehmigt wurde. Genehmigte und abgelehnte Aktionen können auf eine Datei oder über *syslog* protokolliert werden.

Im Bild 1 wird dargestellt, wie der Benutzer *olaf* vom SUDO die Berechtigung für die Änderung des Passworts des Benutzers *arturo* erhält. Die Änderung des *root* Passworts wird jedoch abgelehnt.

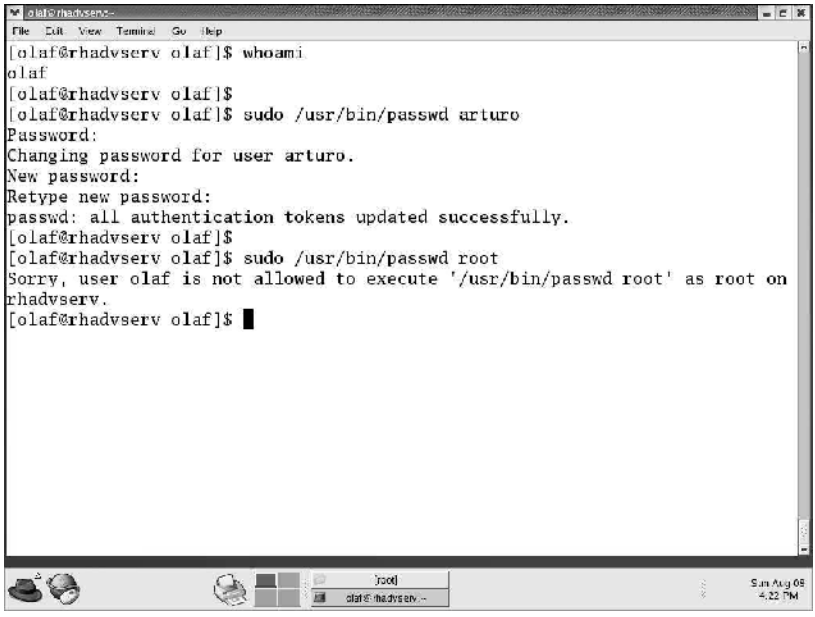


**Abbildung 1:** Ausführung von Befehlen mit SUDO.

Die Zugriffsberechtigungen werden in der Datei */etc/sudoers* konfiguriert. In Bild 2 ist ein Beispiel für eine SUDO Konfiguration abgebildet.

Wie aus Bild 2 zu ersehen ist, kann man mit dem Schlüsselwort *User_Alias* eine Benutzergruppe oder ein Pseudonym für einen Benutzer festlegen, mit dem Schlüsselwort *Run_Alias* kann eine Benutzergruppe oder ein Benutzer für die alternative Identität definiert werden, unter welcher der Befehl ausgeführt werden soll. Das Schlüsselwort

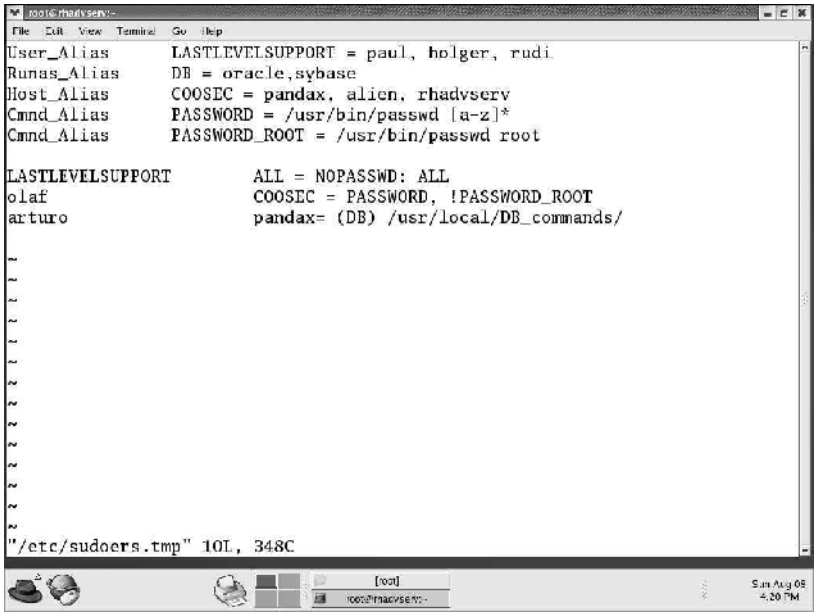


**Abbildung 2:** Beispiel einer SUDO Konfigurationsdatei.

*Host_Alias* wird verwendet, um eine Rechnergruppe oder ein Pseudonym für einen Rechner zu bestimmen. Mit dem Schlüsselwort Cmnd_Alias wird eine Gruppe von Befehlen zusammengefasst oder ein Alias für einen Befehl definiert.

Die vereinfachte Syntax für die Definition einer Berechtigung in SUDO ist:

Benutzername Rechnername = (Alternative Identität) Befehlsname

Die erste Berechtigung in der Konfigurationsdatei aus Bild 2 erlaubt den Mitgliedern der Benutzergruppe *LASTLEVELSUPPORT*, also den Benutzern *paul, holger* und *rudi*, auf allen Rechnern alle Befehle unter der *root*-Identität auszuführen, ohne sich mit dem eigenem Betriebssystempasswort authentifizieren zu müssen. Die zweite Berechtigung erlaubt dem Benutzer *olaf* auf den Rechnern in der Rechnergruppe *COOSE*, also auf den Rechnern *pandax*, *alien* und *rhadvserv*, den Befehl */usr/bin/passwd* auszuführen und damit das Passwort für alle Benutzer bis auf *root* zu ändern. Die dritte Berechtigung erlaubt dem Benutzer *arturo* auf dem Rechner *pandax* unter der Identität der Benutzer *oracle* oder *sybas* (Mitglieder der Benutzergruppe *DB*) alle Befehle im Verzeichnis */usr/local/DB_commands/* auszuführen.

SUDO ist ein einfaches und kostenloses Werkzeug, um die Delegation von *root*-Rechten zu implementieren. SUDO eignet sich bedingt für komplexe Umgebungen, denn SUDO bietet im Standardlieferumfang keine zentrale Administration der Konfigurationsdatei und somit keine zentrale Verwaltung der Security Policy. Für die Verteilung der Konfigurationsdatei kann man auf Standard-UNIX-Tools wie *rdist* zurückgreifen, um nur eine einzige Konfigurationsdatei zu pflegen und diese zu verteilen. Diese Methode ist aus Sicherheits-

gründen nur bedingt zu empfehlen. Ferner kann man mit SUDO durch eine unaufmerksame Erstellung der Sicherheitsrichtlinien *back doors* konfigurieren, denn alle Programme, welche ein *shell escape* ermöglichen, erlauben die Ausführung einer *shell* mit allen Rechten von *root*. Diese *back doors* kann man für *dynamically linked executables* mit dem Flag *NOEXEC* unterbinden. Dieser Flag funktionert bei *statically linked executables* nicht.

Die Protokollierung der ausgeführten Befehle ist mit SUDO entweder über *syslog* oder auf eine Datei möglich. Eine ausführliche Protokollierung aller Tastatureingaben ist mit SUDO nicht möglich; diese fehlende Funktionalität kann ein Ausschlusskriterium für den Einsatz von SUDO auf Systemen mit hohen Sicherheitsanforderungen sein.

## 3 Powerbroker

Powerbroker ist ein kommerzielles Produkt für die Delegation von *root*-Privilegien oder Rechten von anderen Benutzern auf UNIX- und Linux-Systemen der Firma Symark. Die Delegation von Rechten kann über eine Kombination aus

- Benutzernamen oder Benutzergruppen
- Rechnernamen oder Rechnergruppen
- Zeit (Datum und Uhrzeit)
- Programmnamen(Pfad und Checksum)
- Smart Cards
- dem*requesting host* (Rechner, auf dem eine Autorisierungsanfrage gestellt wird)
- dem *executing host* (Rechner, auf dem eine autorisierte Aktion durchgeführt wird)

zusammen gesetzt werden.

Bild 3 zeigt die Interaktion des Benutzers *olaf* mit Powerbroker, um das Passwort des Benutzers *arturo* zu ändern. Der Versuch, das Passwort von *root* zu ändern, wird erwartungsgemäß abgelehnt.

Die Architektur von Powerbroker ist für den Einsatz von Powerbroker in einer verteilten Umgebung konzipiert, diese ist im Bild 4 abgebildet.

Die Architektur von Powerbroker sieht einen Autorisierungsserver *(Master Host)* vor, welcher die Autorisierungsanfragen von *Submit Host* entgegen nimmt und nach Überprüfung der Anfrage in der Security Policy genehmigt oder ablehnt. *Pbmasterd*, der Autorisierungsdaemon, schickt alle Ereignisse zum Speichern an den Protokollierungsserver (*Log Server*).

Der Protokollierungsdaemon *Pblogd* unterscheidet bei der Speicherung der Ereignisse zwischen der Protokollierung der ausgeführten Befehle (*event records*) und der Protokollierung sämtlicher Tastatureingaben und Bildschirmausgaben (*I/O-Log*). *Pblogd* speichert *event records* in der Eventlog Datei und *I/O-Log Records* in *der I/O-Log Datei*. Durch die Protokollierung von Tastatureingaben eignet sich Powerbroker für den Einsatz in Informationssystemen mit hohen Sicherheitsanforderungen, und man kann somit ein revisionsfähiges System implementieren.

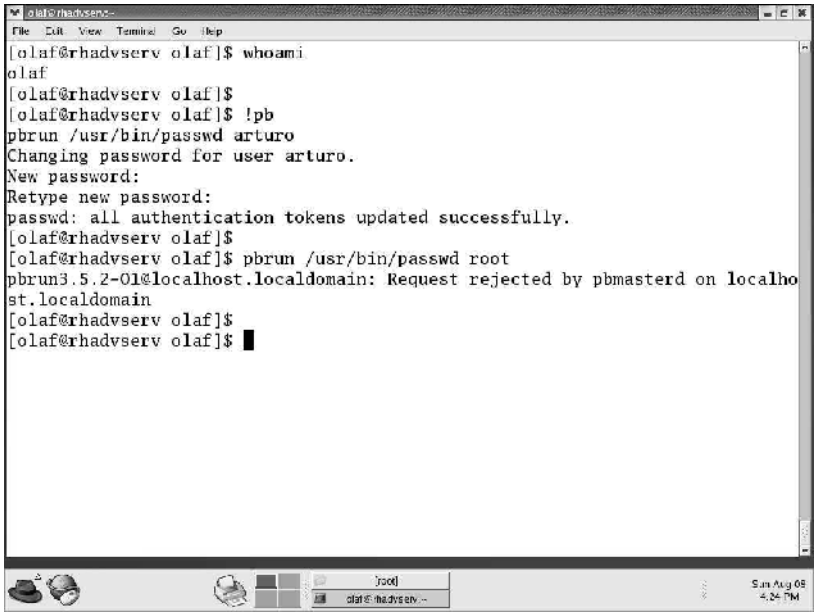


**Abbildung 3:** Interaktion mit Powerbroker.

Powerbroker unterscheidet zwischen Rechnern, welche eine Autorisierungsanfrage einreichen (*Submit Host*) und Rechnern, auf welchen das Programm ausgeführt wird (*Run Host*). Wird diese Trennung konfiguriert, dann leitet *Pbmasterd* genehmigte Autorisierungsanfragen an *pblocald*, den lokalen Daemon auf dem *Run Host*, zur Ausführung weiter. *Pblocald*

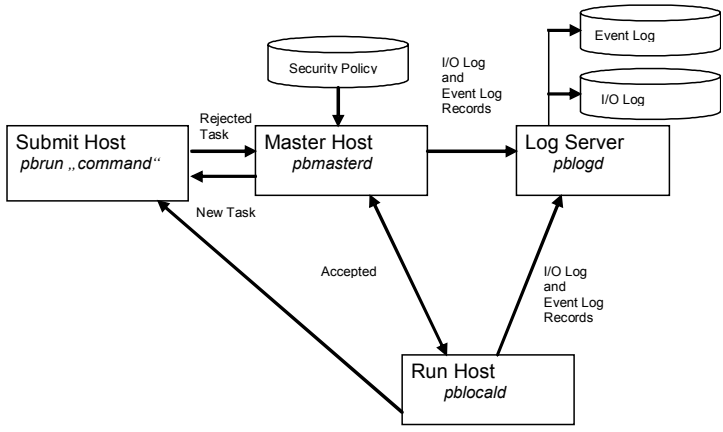


**Abbildung 4:** Architektur von Powerbroker.

protokolliert die ausgeführten Aktionen und generiert *I/O-Logs*- oder *Event-Records* und leitet diese an *pblogd* zur Speicherung weiter.

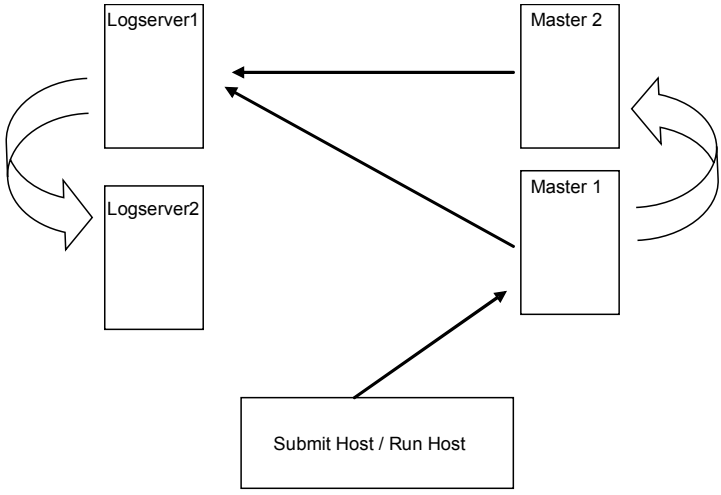


**Abbildung 5:** Schematische Darstellung einer Multimaster- und Multiprotokollierungsserver-Powerbroker-Konfiguration.

Die Ausfallsicherheit der Autorisierungs- und Protokollierungsserver ist in der Powerbroker-Architektur enthalten, die Autorisierungsserver werden in der Konfigurationsdatei */etc/pb.settings* mit den Parametern *submitmasters* sowie *acceptmasters* und die Protokollierungsserver mit dem Parameter *logservers* konfiguriert. Bild 5 zeigt eine Multimaster- und Multiprotokollierung-Serverkonfiguration für Powerbroker.

Die Sicherheitsrichtlinien werden mit Hilfe der Powerbroker Policy Language erstellt, diese Sprache ist eine umfangreiche und leistungsfähige Sprache, welche die Implementierung von exakten und komplexen Sicherheitsrichtlinien ermöglicht. Die Sicherheitsrichtlinien werden in der Datei */etc/pb.conf* konfiguriert. Bild 6 zeigt ein Beispiel einer Powerbroker Security Policy.

In diesem Beispiel wird dem Benutzer *olaf* die Genehmigung erteilt, das Passwort aller Benutzer zu ändern. *olaf* hat keine Genehmigung, das Passwort von *root* zu ändern; der Versuch, das Passwort von *root* zu ändern, wird abgelehnt.

Powerbroker ermöglicht die effektive Kontrolle der Ausführung von Third-Party-Applikationen und Skripten durch die Parametrisierung des Programmaufrufs mit speziellen Parametern für die Ausführung von executables. Powerbroker liefert im Standardlieferumfang sichere Versionen der Standard-UNIX-Programme, die ein *Shell scape* ermöglichen (vipb als Ersatz für vi). In der Powerbroker Security Policy werden diese sicheren Programme aufgerufen. Wird z. B. /usr/local/bin/pbvi statt /usr/bin/vi aufgerufen, ist die Gefahr einer *back door* gebannt.

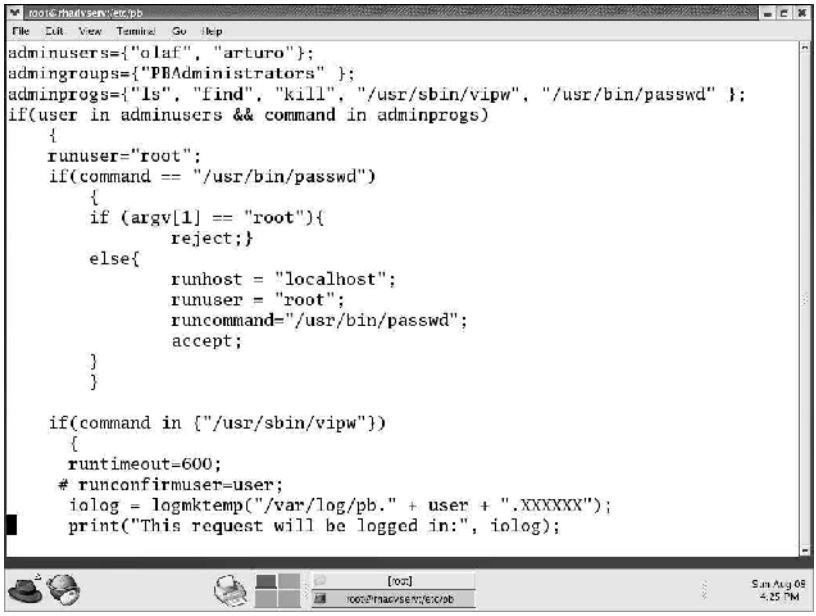


**Abbildung 6:** Beispiel einer Powerbroker-Sicherheitsrichtlinie.

Powerbroker ist für den Einsatz in großen und komplexen Umgebungen sehr gut geeignet, denn Powerbroker unterstützt das zentrale Management der Security Policy und eine ausfallsichere Konfiguration. Es ist möglich mehrere Autorisierungsserver (Multimaster) und mehrere Protokollierungsserver (Multilogserver) zu konfigurieren. Ein weiterer Vorteil von Powerbroker ist die Unterstützung von LDAP für die Abfrage der Security Policy. In einer LDAP-Konfiguration werden die Benutzerkonfiguration und die Security Policy auf einem LDAP-Server gespeichert und somit an einer einzigen Stelle zentral verwaltet. Das bedeutet eine deutliche Verbesserung der Sicherheit gegenüber Konfigurationen, in welchen die Security Policy auf alle Systeme (z. B. Mit *rdist*) verteilt wird. Powerbroker bietet eine ausführliche Protokollierung der genehmigten und abgelehnten Aktionen über den Powerbroker-Protokollierungsserver.

## 4 Identity Management

SUDO und Powerbroker stellen eine Lösung für die Delegation von *root*-Privilegien auf UNIX- und Linux-Systemen dar. Eine effiziente Delegation von *root*-Rechten in komplexen Umgebungen setzt voraus, dass unternehmensweit jeder UNIX *account* einem Mitarbeiter eindeutig zugeordnet werden kann. Diese Zuordnung kann man durch den Einsatz von Identity Management-Architekturen erzielen.

In diesem Abschnitt erläutern wir zuerst die Grundlagen von Identity Management, anschließend beschreiben wir einen Ansatz für die Integration von Powerbroker mit Select

Access und Select Identity (Identity Management-Lösungen von Hewlett Packard). Zuletzt stellen wir die Vorteile der Integration dar.

### 4.1 Warum Identity Management?

In den letzten Jahren haben die Komplexität, die geographische Verteilung und die Anzahl der Datenquellen von Infrastrukturen für Informationssysteme rasant zugenommen. Der Aufwand für die Pflege von Anwenderdaten, für die Verwaltung von Applikationen und die Anforderungen an die Sicherheit dieser Systeme ist daher enorm gestiegen.

Die Meta Group hat in der Studie „The Value of Identity Management“ aus dem Jahr 2002 herausgefunden, dass

- in großen Unternehmen im Schnitt 68 interne und 12 externe Datenspeicher verwaltet werden
- jeder Anwender Informationen auf durchschnittlich 22 Datenspeichern pflegt
- 15% jährliche Änderungsrate der Anwenderdaten 29% der gesamten IT-Ressourcen verbrauchen
- 45% der Help Desk-Anrufe sich auf Passwortprobleme beziehen

Die Anforderungen an die Sicherheit von kritischen Systemen sind gestiegen, denn die Anwender müssen Passwörter für eine Vielzahl von unterschiedlichen Systemen verwalten. Diese Vielzahl von Passwörtern führt dazu, dass der Anwender sich entweder die Passwörter auf Zettel aufschreibt oder sich für einfache Passwörter entscheidet, welche wiederum leicht zu knacken sind.

Der Ansatz des Identity Management ist die zentrale und konsolidierte Verwaltung von Identitätsinformationen. Diese Informationen beschreiben die Merkmale und Eigenschaften, welche die eindeutige Festlegung einer Identität ermöglichen.

Im Identity Management-Kontext versteht man unter Identität in der Regel die Identität einer Person. Eine Identität kann aber auch eine Organisationseinheit, eine Anwendung oder einen Dienst beschreiben.

Die Architektur von Identiy Management umfasst: Den globalen Verzeichnisdienst (Global Data Repository) Die verzeichnisbasierte Anwenderadministration und Identitätssynchronisation Das User Provisioning

### 4.2 Das Access Management

Die zentrale Komponente der Identity Management-Architektur ist der globale Verzeichnisdienst. Im globalen Verzeichnisdienst wird die Information aller Identitäten im Unternehmen zentral und konsolidiert verwaltet. Dadurch wird sichergestellt, dass die Eigenschaften und Merkmale einer Identität unternehmensweit eindeutig sind. Der globale Verzeichnisdienst kann als Enterprise Directory, als Metadirectory oder als Virtual Directory implementiert werden. Das Enterprise Directory verfolgt den Ansatz, alle Informationen an einer zentralen Stelle, am Global Data Pool, zu konsolidieren und einen gemeinsamen Datenpool für alle Anwendungen zu schaffen. Beim Metadirectory wird kein Global Data

Pool implementiert, sondern die einzelnen Directories werden synchronisiert, wobei einige Implementationen einen Global Data Pool für die Synchronisation verwenden. Das Virtual Directory verzichtet auf den Global Data Pool und geht über die Idee des Metadirectory hinaus, indem es eine einheitliche Schnittstelle zu allen Datenquellen im Unternehmen anbietet und die Informationen auf allen Datenquellen dynamisch abgleicht. Die Anwenderidentitäten werden dann mit Hilfe des globalen Verzeichnisdienstes verwaltet und synchronisiert.

Das User Provisioning liefert einen Mehrwert für den Anwender, indem es die Anmeldung auf verschiedenen Systemen durch Single Sign On vereinfacht, dem Anwender die Pflege der eigenen Informationen ermöglicht und dem Anwender die Berechtigung gibt, Routineaufgaben (z. B. Systemregistrierung, Passwort-Reset) selbst auszuführen. Die Implementation von User Provisioning gibt dem Anwender mehr Flexibilität und entlastet die Systemadministratoren.

Die Realisierung einer zentralen Zugriffskontrolle (Access Management) setzt voraus, dass ein globaler Verzeichnisdienst bereits existiert. Nur dann ist gewährleistet, dass eine Identität unternehmensweit eindeutig ist.

Die Steuerung des Zugriffs auf Anwendungen ist in einer homogenen Umgebung in der Regel unproblematisch und wird über ein zentrales Verzeichnis, z. B. Active Directory realisiert. Der Zugriff auf Anwendungen in einer heterogenen Umgebung kann durch den Einsatz von einem LDAP-Directory realisiert werden, falls die Anwendungen die LDAP-Schnittstelle unterstützen. Diese Funktionalität wird benötigt, um SUDO oder Powerbroker in ein Identity Management System zu integrieren.

### 4.3 OpenView Select Access

OpenView Select Access ist ein zentrales Zugriffskontrollsystem, mit welchem unternehmensweite Sicherheitsrichtlinien und rollenbasierte Zugriffsberechtigungen auf Unternehmensressourcen konfiguriert werden können.

OpenView Select Access unterstützt:

- Single Sign On
- Definition von Anwenderprofilen
- Verwaltung von Anwenderpasswörter und -profile
- Delegation von Administrationsaufgaben
- End-to-End Auditing
- Autodiscovery von Ressourcen im Unternehmen

Die Identität eines Anwenders wird durch ein Profil beschrieben. Ein Profil besteht aus einer Reihe von Attributen, welche den Anwender und die dazu gehörigen Berechtigungen oder Rollen beschreiben. Diese Rollen werden dazu genutzt, in einem rollenbasierten Zugriffskontrollsystem die Berechtigungen des Benutzers zu ermitteln. Die Attribute können als Umgebungsvariablen exportiert werden. Die Applikationen können dann auf diese Variablen zugreifen und die Berechtigungen des Anwenders ermitteln.

Die Komponenten von Select Access sind Admin Server, Secure Audit Server, Policy Validator, Enforcer Plug-In, SAML Server und LDAP Directory. Bild 7 zeigt die Architektur von OpenView Select Access. Die Benutzerinformationen und die Sicherheitsrichtlinien werden auf einem LDAP-Server gespeichert.

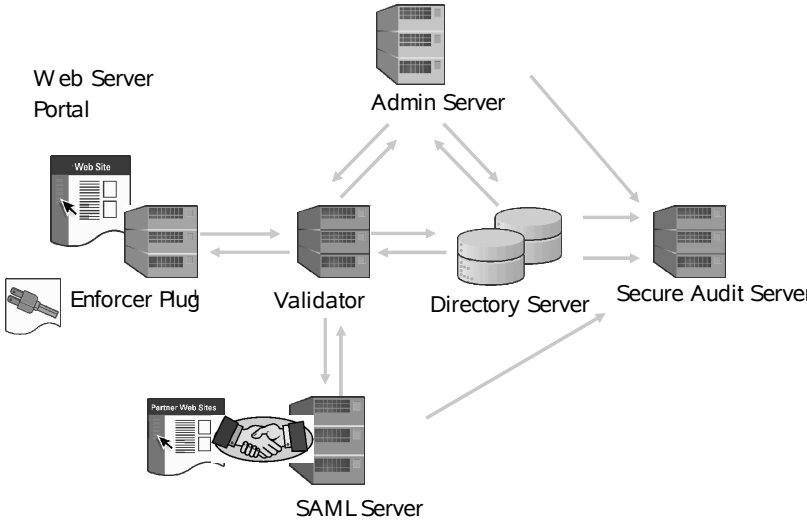


**Abbildung 7:** Architekturbild von OpenView Select Access (Quelle: HP OpenView Select Access Installation Guide).

### 4.4 OpenView Select Identity

OpenView Select Identity ist ein Identity Management-Produkt von Hewlett Packard, welches eine Lösung für das User Provisioning und Identitätssynchronisation darstellt. Das Sicherheitsmodell von OpenView Select Identity basiert auf dem Modell des Contextual Identity Managements, welche eine Erweiterung des Role Based Access Control Model des National Institute of Standards and Technology (NIST) ist.

Mit OpenView Select Identity werden Anwenderinformationen zentral verwaltet (Benutzer anlegen, verändern und entfernen), Passwörter im LDAP-Verzeichnis administriert oder auf den verschiedenen Systemen synchronisiert und Self Service-Funktionen für den Anwender realisiert.

Die Identitätsinformationen der Anwender werden mit Hilfe von Java Connector Architecture (JCA) Konnektoren zwischen den verschieden Systemen über ein Virtual Directory synchronisiert. Diese Funktion ist sehr wichtig für die Integration von SUDO oder Powerbroker mit OpenView Select Identity und somit für die Integration mit einem Identity Management-System, denn über diese Konnektoren werden das Home Directory des Benutzers und die benutzerspezifische Konfiguration (*Login*-Skripte, Umgebungsvariablen) auf den verschieden UNIX- und Linux-Systemen angelegt.

## 5 Integration von Powerbroker in ein Identity Management System

Für die Integration von Powerbroker in ein Identity Management sind folgende Schritte nötig:

- Konfiguration von Powerbroker für die Unterstützung von LDAP
- Definition einer zentralen Powerbroker-Sicherheitsrichtlinie, die ein rollenbasiertes Zugriffkontrollsystem unterstützt
- Konfiguration der UNIX- und Linux-Rechner für die Betriebssystemanmeldung über LDAP
- Integration der UNIX- und Linux-Rechner in das User Provisioning des Identity Management Systems
- Integration von Powerbroker in das unternehmensweite rollenbasierte Zugriffskontrollsystem des Identity Management Systems

Die Integration von Powerbroker in ein Identity Management System wird in diesem Abschnitt anhand der Einbindung von Powerbroker in die Identity Management-Produkte von Hewlett Packard gezeigt. Diese erfolgt im Wesentlichen über die Anbindung von Powerbroker an das LDAP Directory. In diesem Beispiel wird OpenView Select Identity in erster Linie für das User Provisioning, also für das Anlegen von Benutzer-*accounts* auf dem LDAP Server, und für das Anlegen von *Home-Verzeichnissen* auf den UNIX- und Linux-Rechnern eingesetzt. OpenView Select Access wird vor allem für die Realisierung des unternehmensweiten rollenbasierten Zugriffskontrollsystems eingesetzt. Bild 8 zeigt schematisch die Integration der unterschiedlichen Produkte über das LDAP Directory.

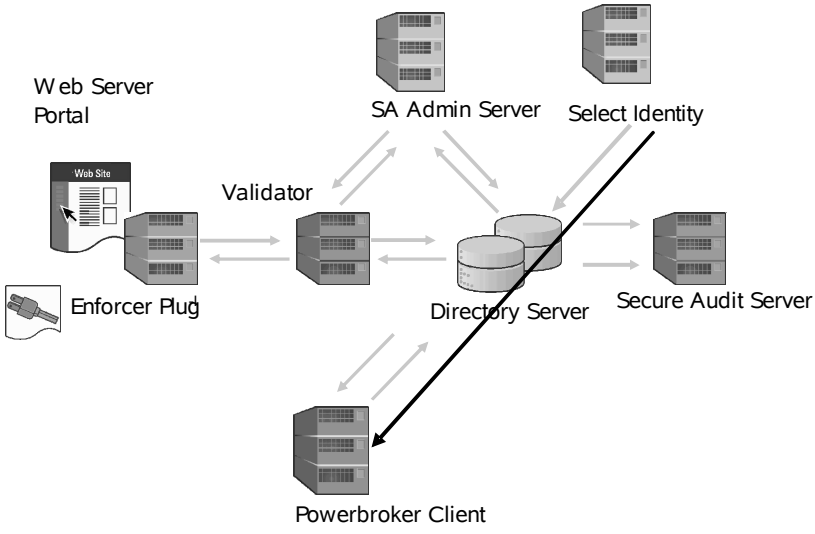


**Abbildung 8:** Integration von Powerbroker in Select Access und Select Identity über LDAP.

Um Powerbroker-Sicherheitsrichtlinien in einen LDAP-Verzeichnis speichern zu können, muss das Powerbroker-Schema auf dem LDAP-Server definiert werden. Dieses Schema legt Attribute und Objekte für die Powerbroker-Sicherheitsrichtlinie fest. Mit dieser Erweiterung des LDAP-Schemas kann man ein rollenbasiertes Zugriffkontrollsystem aufbauen, welches sich leicht in das unternehmensweite rollenbasierte Zugriffskontrollsystem von OpenView Select Access integrieren lässt.

Auf den Rechnern, auf welchen *root*-Rechte mit Hilfe von Powerbroker delegiert werden sollen, wird eine lokale Powerbroker-Sicherheitsrichtlinie konfiguriert, welche die Anwenderberechtigungen auf der zentralen Sicherheitsrichtlinie (die im LDAP-Verzeichnis gespeichert ist) überprüft, um eine lokale Autorisierungsanfrage zu genehmigen oder abzulehnen. Zusätzlich muss die Betriebssystemanmeldung über LDAP konfiguriert werden.

Die Benutzer werden mit OpenView Select Identity im LDAP-Verzeichnis angelegt, OpenView Select Identity legt über die JCA Konnektoren das Home Directory des Benutzers auf den entsprechenden Linux- und UNIX-Systeme an.

Die Zuordnung der Rollen zu einem Benutzer erfolgt über das unternehmensweite, rollenbasierte Zugriffsteuerungssystem, also über OpenView Select Access. Die zentrale Powerbroker-Sicherheitsrichtlinie ist weitgehend statisch und muss nur dann verändert werden, wenn das unternehmensweite, rollenbasierte Zugriffskontrollsystem um neue Rollen erweitert wird und diese neuen Rollen für die Delegation von *root*-Rechten an UNIX- oder Linux-Benutzer relevant sind.

Diese Integration zeigt, dass eine sichere und effiziente Delegation von *root*-Rechten an Anwender in komplexen Umgebungen nur dann realisiert werden kann, wenn die Anwenderidentitäten in der gesamten Umgebung eindeutig sind und die Berechtigungen in Verbindung mit einem rollenbasierten Zugriffkontrollsystem vergeben werden. Die Vorteile dieser Integration sind:

- Reduktion des Pflegeaufwands für die Powerbroker-Sicherheitsrichtlinie
- Elimination des mehrfachen Aufwands für die Pflege der Benutzerberechtigungen auf verschiedenen Zugriffskontrollsystemen (UNIX, Windows, Anwendungen) und somit Kostenreduktion für die Benutzerverwaltung
- Erhöhung der Sicherheit des Gesamtsystems durch Sicherstellung der eindeutigen Zuordnung von Berechtigungen zu einer Identität und durch die zentrale Pflege der Sicherheitsrichtlinien des Unternehmens.

## 6 Zusammenfassung

Die Delegation von *root*-Rechten an UNIX- und Linux-Anwender kann man mit SUDO in übersichtlichen Umgebungen gut realisieren. Komplexe Umgebungen, die höhere Anforderungen an die Sicherheit und Verfügbarkeit stellen, erfordern Lösungen, die Ausfallsicherheit, ausführliche Protokollierung und zentrales Management der Sicherheitsrichtlinien anbieten. Powerbroker erfüllt diese Anforderungen.

Die Delegation von *root*-Rechten an UNIX-Anwender kann in Umgebungen, die sehr hohe Sicherheits- und Verfügbarkeitsanforderungen haben, geographisch verteilt sind und ge-

setzliche Richtlinien erfüllen müssen, nicht losgelöst vom unternehmensweiten Zugriffskontrollsystem implementiert und betrieben werden. Denn diese Umgebungen erfordern eine eindeutige Zuordnung von Berechtigungen zu Identitäten und somit zu Anwendern, sowie eine ausführliche Protokollierung der durchgeführten Aktionen, um die Ursachen von bestimmten Ereignissen abgrenzen zu können. Diese Anforderungen können mit der Integration von einem Werkzeug für die Delegation von *root*-Rechten an ein Identity Management System erfüllt werden.

In diesem Beitrag haben wir gezeigt, wie durch die Integration von Powerbroker in OpenView Select Access und OpenView Select Identity die Anforderungen von komplexen Umgebungen an die Delegation von *root*-Rechten an UNIX- und Linux-Anwendern erfüllt werden können.