Authentifizierung mit Oracle Identity Management Infrastructure, Oracle Access Manager und WebGate

Michael Pergande und May Schneider, PROMATIS software GmbH

Gerade im Umfeld von OC4J-Applikationen bietet es sich an, die Authentifizierung vollständig an die Oracle Identity-Management-Komponenten auszulagern und transparent für die Applikationen zu gestalten. In diesem Artikel wird anhand eines typischen Szenarios die Vorgehensweise einer solchen Auslagerung im Hinblick auf die Server-Architektur, die zu installierenden Produkte und deren Konfiguration erläutert. Dabei wird insbesondere auf eine im Internet erreichbare Konfiguration mit einem separaten http-Server in einer neutralen Zone eingegangen.

Die mit dem Aufruf einer geschützten Web-Applikation (OC4J) verbundene Authentifizierung soll vollständig an die Komponenten des Oracle Identity Managements ausgelagert werden. Der Zugriff auf die Ressource soll dabei sowohl intern als auch extern erfolgen können. Bei beiden Zugriffsformen ist zu gewährleisten, dass der Authentifizierungsmechanismus nicht umgangen werden kann. Kritische Strecken sind durch Verschlüsselung geschützt.

Lösungskonzept

Die Authentifizierung eines Benutzers, der eine geschützte Anwendung anfordert, erfolgt in drei Phasen.

- In Phase 1 erfolgt die Überprüfung der Zugriffsmodalitäten. Dafür kommt ein Server, der über ein WebGate-Plugin verfügt, zum Einsatz. Dieses Plugin ermöglicht es, den Ressourcen-Aufruf abzufangen und den Benutzer zur Authentifizierungskomponente umzuleiten, wobei dies transparent für die Anwendung erfolgt
- In Phase 2 erfolgt die Authentifizierung des Benutzers mit den zuvor erhaltenen Credentials der Login-Seite. Dazu werden diese an den Server, der die Oracle Identity-Management-Komponenten und eine Datenbank beinhaltet, weitergeleitet und verifiziert
- In Phase 3 wird die Applikation freigegeben, sofern der Benutzer über die benötigten Berechtigungen verfügt. Im nächsten Schritt müssen bei Bedarf Autorisierungsschritte erfolgen. Dies kann über den Policy Server des Access Managers oder applikationsintern erfolgen. Dies wird in diesem Artikel nicht weiter behandelt.

Hardware- und Software-Architektur

Zur Umsetzung des Konzepts sind auf dem Server, der die Datenbank und das Oracle Identity Management enthält (H80003), die Komponenten Identity Management Infrastructure und Access Manager installiert (siehe Abbildung 1). Durch den Einsatz der Identity Management Infrastructure verfügt der Server über eine Datenbank mit dem Oracle Internet-Directory-Schema (OID) und einen LDAP-Dienst, mit dessen Hilfe die Anfragen vom Access Manager bearbeitet werden können. Dort liegen zentral die Benutzerinfor-

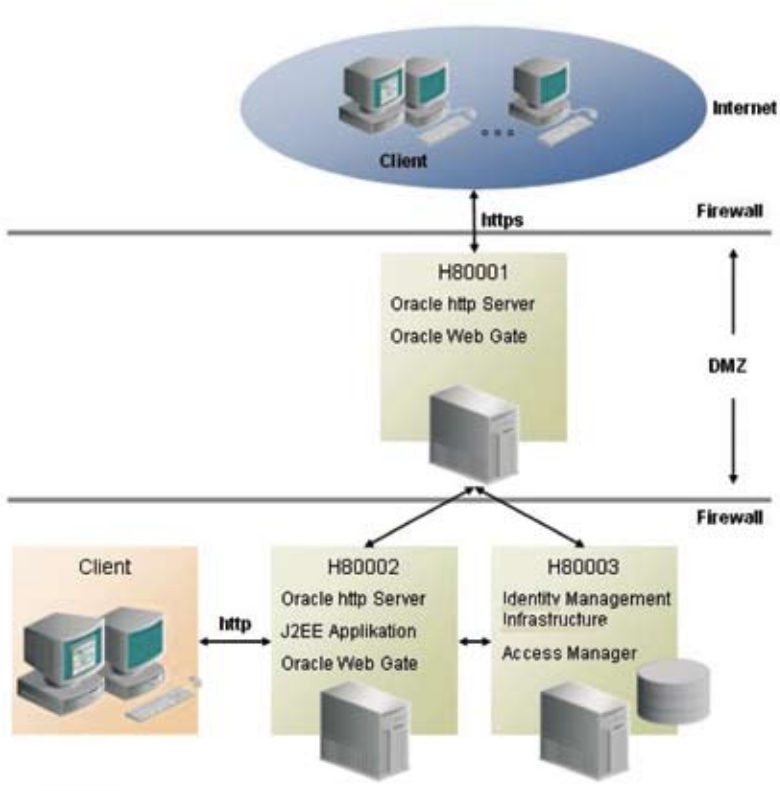

Abbildung 1: Übersicht der Hard- und Software-Komponenten

mationen, auf die man über das LDAP-Protokoll zugreift.

Der Access Manager stellt einen Identity- und einen Access-Server zur Verfügung. Mit dem Identity-Server lassen sich unter anderem Benutzer und Gruppen verwalten und Stellvertreter-Funktionen festlegen. Die Daten speichert der Identity Server ebenfalls im OID.

Die Hauptaufgabe des Authentifizierungsmechanismus übernimmt der Access-Server. Er empfängt die vom Access-Client gesendeten Credentials, überprüft mithilfe der Datenbank deren Zulässigkeit und gibt gegebenenfalls die Ressource frei. Dies resultiert im Setzen von Cookies auf dem Client, in denen die erfolgte Authentifizierung festgehalten ist.

Um Zugriffe auf die geschützte Anwendung abzufangen, wird WebGate als Access-Client verwendet. Da sowohl der interne als auch der externe Zugriff über den Access-Manager abgewickelt werden soll, werden zwei WebGates benötigt. Das WebGate für den externen Zugriff liegt auf einem separaten Server (H80001), der sich in der DMZ befindet. Dieser betreibt lediglich einen Apache HTTP-Server mit WebGate und leitet alle Requests für Seiten, die extern zugreifbar sein sollen, nach Authentifizierung an den eigentlichen Applikationsserver (H80002) weiter. Für den internen Zugriff wird WebGate auf demselben Server, der auch die geschützte Anwendung enthält, integriert. Um die Datensicherheit bei externem Zugriff zu gewährleisten, erfolgt die Kommunikation zwischen Internet Client und WebGate über https.

Konfiguration

Im ersten Schritt wird der Server, auf dem die Oracle Identity-Management-Komponenten enthalten sein sollen, eingerichtet. Dies beinhaltet die Installation der Datenbank, der Identity-Management-Infrastructure-Komponenten und des Access-Managers.

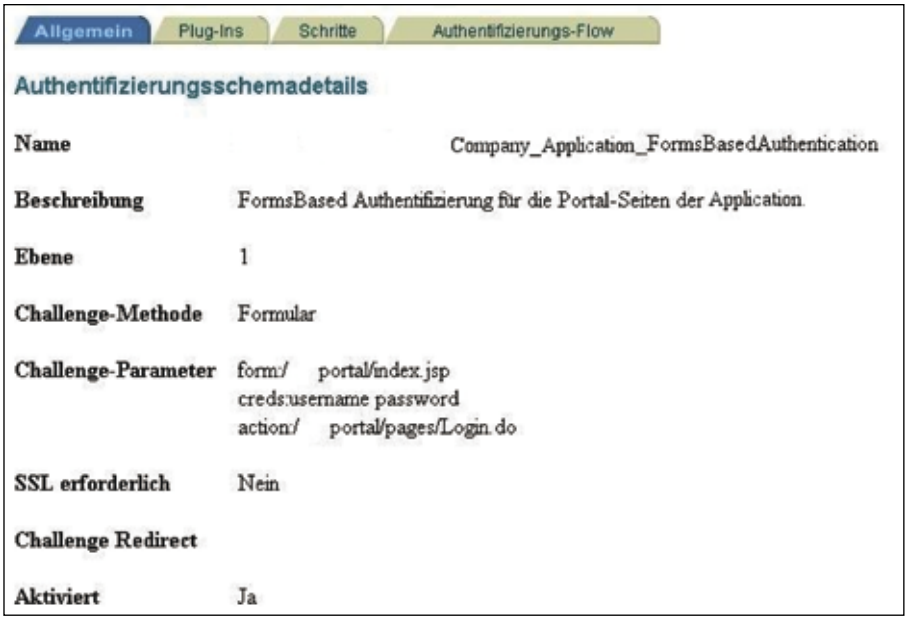

Abbildung 2: Authentifizierungsschema

Abbildung 3: Authentifizierungsschritte

Vor der Konfiguration des Access-Managers wird für die zu schützende Applikation eine Login-Seite entwickelt, mit deren Hilfe sich die Benutzer-Credentials für die Authentifizierung extrahieren lassen. Damit die Credentials ausgelesen werden können, muss die Login-Seite dem Access Manager innerhalb eines Authentifizierungsschemas bekannt gemacht werden (siehe Abbildung 2).

Der Abschnitt Challenge-Parameter ist dabei besonders wichtig. Er enthält den Namen der Login-Seite sowie die Namen der Felder, aus denen die zu prüfenden Credentials gelesen werden. Neben den Datenquellen werden im Authentifizierungsschema auch die einzelnen Schritte des Authentifizierungsmechanismus angegeben (siehe Abbildung 3).

Die Plugin-Parameter enthalten dabei die Knoten im OID, in denen die zur Authentifizierung notwendigen Informationen stehen. Zusätzlich werden dem Access-Manager die Access-Gates (H80001, H80002), die für das Abfangen der Requests auf die zu schützende Ressource verantwortlich sind, bekannt gemacht.

Bereits bei der Installation des Access-Managers muss eine LDAP-Quelle angegeben werden. Diese beinhaltet die Daten des Directory-Servers und die Profile des LDAP-Directory-Servers (siehe Abbildungen 4 und 5).

Nach der Konfiguration des Access-Managers müssen auf den Access-Gate-Servern (H80001, H80002) die WebGate-Plugins integriert werden. Dazu gehört auch das Festlegen der Kommunikationsprotokolle zwischen Client und WebGate. Während zwischen einem internen Client und WebGate über http kommuniziert werden kann, bedarf es zwischen einem externen Client und WebGate einer zusätzlichen Verschlüsselung, um die kritische Strecke vom Internet in die DMZ zu überbrücken. Dies geschieht durch den Einsatz von SSL. Es sollte aber trotzdem ermöglicht werden, den initialen Request per http zu senden; in diesem Fall wird automatisch auf https gewechselt. Spricht man den Port des Servers in der DMZ an, wird dieser auf einen weiteren Port verzweigt, der die

Daten via SSL verschlüsselt überträgt. Dazu sind in der Apache-Konfiguration Virtual-Hosts und entsprechende Regeln definiert. Relevant sind dabei die Konfigurationsdateien httpd.conf (siehe Abbildung 6) und ssl.conf (siehe Abbildung 7). In der Datei ssl.conf werden auch die Zertifikats-Informationen für die SSL-Verbindung konfiguriert. Es wird dort auf ein Wallet verwiesen, das mittels des Oracle Wallet-Managers erstellt wurde.

Wie bereits erwähnt, wird WebGate für den externen Zugriff auf einem Server in der DMZ und für den internen Zugriff auf dem Applikations-Server integriert. Bei einem externen Zugriff auf die Ressource muss somit sichergestellt sein, dass das WebGate auf dem Applikationsserver nicht erneut den Request abfängt. Dies wird erreicht, indem man den Applikations-Server (H80002) über zwei Ports anspricht. Alle Requests vom Server in der DMZ kommen über Port 7778 und müssen nicht erneut authentifiziert werden. Alle internen Requests kommen über Port 7777 und werden von WebGate authentifiziert. Durch entsprechende Netzwerk-Konfiguration muss sichergestellt sein, dass keine internen Requests über Port 7778 ankommen können.

Erfolgt der Zugriff über den Server in der DMZ, wird die Ressource immer

Directory-Server

Rechner	h80003 intern.company .de
Port-Nummer	389
Root-DN	cn=orcladmin
Root-Kennwort	*<Nicht angezeigt>*
Suchbasis	dc=intern,dc=Company ,dc=de
Konfigurationsbasis	o=Oblix,cn=OAM Identity,cn=Products,cn=OracleContext
Security-Modus des Directory-Servers	Open
Unabhängige Suchbasis	

Abbildung 4: Datenbank-Profil – Directory Server

Profile für LDAP-Directory-Server konfigurieren

	Name	Namespace	Primäre Server	Sekundäre Server
☐	oblixConfig-IdentityServer	o=Oblix,cn=OAM Identity,cn=Products,cn=OracleContext	default	
☐	default-IdentityServer	dc=intern,dc=Company ,dc=de	default	
☐	AccessManager_setup_user_profile	dc=intern,dc=Company ,dc=de	default	
☐	AccessManager_setup_oblix_profile	o=Oblix,cn=OAM Identity,cn=Products,cn=OracleContext	default	
☐	AccessManager_setup_policy_profile	o=Oblix,cn=OAM Policy,cn=Products,cn=OracleContext	default	
☐	AccessServer_default_user_profile	dc=intern,dc=Company ,dc=de	default	

Abbildung 5: Datenbank-Profil – Profile LDAP-Directory-Server

```
# This port is used when starting without SSL
Port 80
Listen 80
NameVirtualHost www.company-portal.de:80
<VirtualHost www.company-portal.de:80>
    ServerName www.company-portal.de
    Port 443
    SimulateHttps on
    RewriteEngine on
    RewriteOptions inherit
    RewriteRule ^/(.*)$ https://www.company-portal.de/$1
[R,L]</VirtualHost>
```

Abbildung 6: Auszug aus der Datei httpd.conf von h80001

```
Listen 443
<VirtualHost www.company-portal.de:443>
     # SSL Engine Switch:
     # Enable/Disable SSL for this virtual host.
     SSL Engine on
     ServerName www.company-portal.de
     Port 443
     ...
     SSLWallet file:/home/oracle/wallets/production
     ...
     ProxyPass /application/
     http://H80002.intern.company.de:7778/application/
      ProxyPassReverse /application/
     http://H80001.intern.company.de:7778/application/
     ...
     SimulateHttps on
      AddCertHeader HTTPS
      RewriteEngine on
      RewriteOptions inherit
</VirtualHost>
```

Abbildung 7: Auszug aus der Datei ssl.conf von h80001

freigegeben. Bei einem Zugriff über den Applikations-Server wird der Request abgefangen und an den Access-Manager weitergeleitet (siehe Abbildung 8).

Fazit

Bei der vorgestellten Methode zur Auslagerung der Authentifizierung in die Komponenten des Oracle Identity-Managements handelt es sich um eine von vielen Lösungsvarianten. Mit dieser Architektur und Konfiguration ist es möglich, die Authentifizierung und bei Bedarf auch die Autorisierung ausschließlich mit den Komponenten von Oracle Identity-Management komplett transparent für die zu schützenden Applikation(en) abzuwickeln, dabei zusätzlich relevante Strecken verschlüsselt zu übertragen und unterschiedliche Zugriffswege für interne und externe Benutzer zu verwenden. Dazu sind nach Implementierung der Applikation ohne Berücksichtigung der Authentifizierung nur wenige Installations- und Konfigurationsschritte durchzuführen.

```
# This port is used when starting without SSL
Port 7777
Listen 7777
Listen 7778
NameVirtualHost *:7778
NameVirtualHost *:7777
<VirtualHost *:7778>
      ServerName www.company-portal.de
      Port 80
</VirtualHost>
<VirtualHost *:7777>
      ServerName H80002.intern.company.de
      Port 7777
      <LocationMatch "/*">
             AuthType Oblix
             Require valid-user
      </LocationMatch>
</VirtualHost>
```

Abbildung 8: Auszug aus der Datei httpd.conf des Applikations-Servers

Kontakte:
Michael Pergande
michael.pergande@promatis.de
May Schneider
may.schneider@promatis.de