Magnet™ Mobile App Server Release Notes 2.0

Revision A

MAGNET SYSTEMS PROVIDES NO WARRANTY WITH REGARD TO THIS MANUAL, THE SOFTWARE, OR OTHER INFORMATION CONTAINED HEREIN AND HEREBY EXPRESSLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE WITH REGARD TO THIS MANUAL, THE SOFTWARE, OR SUCH OTHER INFORMATION. IN NO EVENT SHALL MAGNET SYSTEMS, INC. BE LIABLE FOR ANY INCIDENTAL, CONSEQUENTIAL, OR SPECIAL DAMAGES, WHETHER BASED ON TORT, CONTRACT, OR OTHERWISE, ARISING OUT OF OR IN CONNECTION WITH THIS MANUAL, THE SOFTWARE, OR OTHER INFORMATION CONTAINED HEREIN OR THE USE THEREOF.

Magnet Systems, Inc.

435 Tasso Street
Suite 100
Palo Alto, CA
94301

650–329–5904

info@magnet.com

Contents

Overview

The Magnet™ Mobile App Server enables enterprises to easily build applications and deploy in the cloud for their employees and their customers while reducing the time to market, cost and complexity of development. The Magnet Mobile App Server offers a solution that gives mobile users rich and personally relevant experiences, as well as the capabilities of their preferred mobile devices.

Magnet Mobile App Server

The Magnet Mobile App Server provides basic constructs for creating and manipulating app objects spanning the server and mobile app contexts. The Magnet Mobile App Server includes the Magnet Mobile Enterprise Server (MES) and Magnet Mobile Server.

The diagram on the next page shows the Magnet Mobile App Server components and their relationships.

Magnet Mobile Enterprise Server

MES is an application server that runs in the cloud and business logic is constructed in the Developer Factory. Instances of MES are deployed in the cloud, and may contain a blend of off-the-shelf (OTS) and custom controllers.

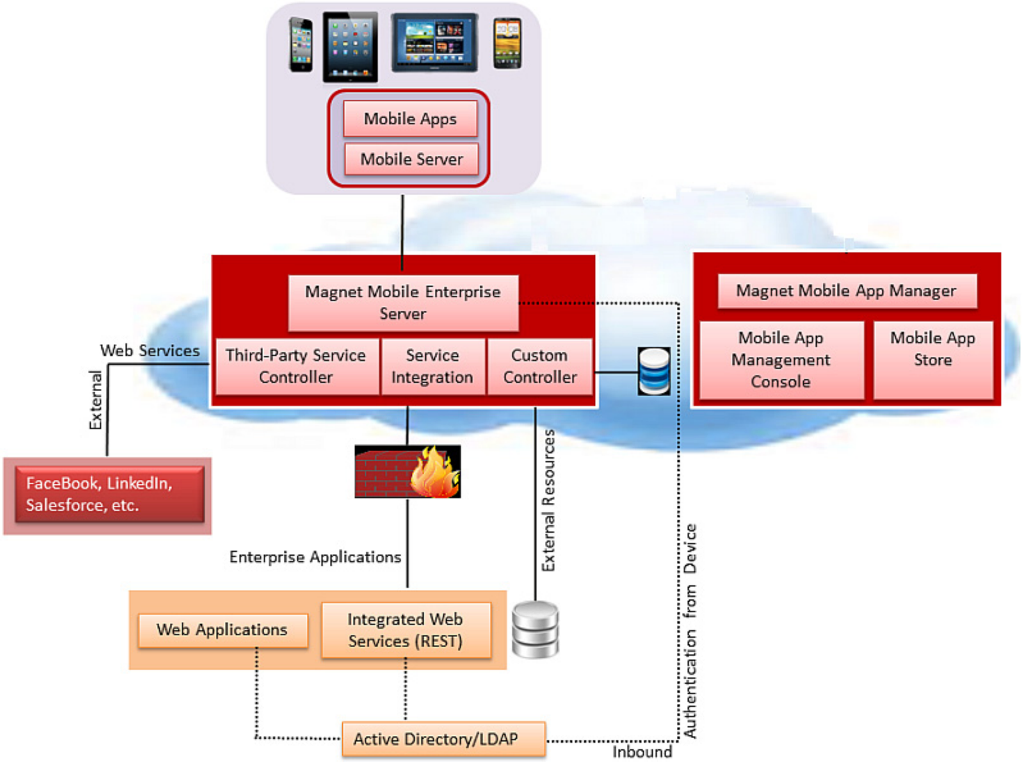
An executable JAR file contains all controllers required for a specific deployment. Instances of the Magnet Mobile Enterprise Server are deployed in the Amazon cloud —a virtual private cloud that is managed by customers.

All configurable properties for all controllers and the Magnet Mobile Enterprise Server are outside from the executable JAR file. Any configuration properties can be configured without having to recompile the JAR file, which reduces the possibility of injecting errors in the JAR File.

Magnet Mobile Server

The Magnet Mobile Server is installed on a device and only one Magnet Mobile Server is installed on a device at any time. The Magnet Mobile Server can be deployed either as a separate process (on Android only) or embedded within the app (on both iOS and Android). The Magnet Mobile Server provides user, device, and app management, security, and communication on the mobile device.

For security purposes, only applications installed by the Magnet Mobile Server is allowed to communicate through the Magnet Mobile Server. The Magnet Mobile Server manages a single sign-on (SSO) for the device when a user is authenticated. Through an integration with a configured LDAP server, the Magnet Mobile Server manages credentials and authenticates on the user's behalf for each app session.


Mobile Apps
Mobile Server
Magnet Mobile Enterprise Server
Third-Party Service Controller
Service Integration
Custom Controller
Magnet Mobile App Manager
Mobile App Management Console
Mobile App Store
Web Services
External
FaceBook, LinkedIn, Salesforce, etc.
Enterprise Applications
External Resources
Authentication from Device
Web Applications
Integrated Web Services (REST)
Active Directory/LDAP
Inbound

Security

The Magnet Mobile App Server supports industry standards for secure access to the public and private back-end services:

- Basic authentication is used for client to MES and MES to Web service.
- WS-security is used for Magnet MES to Web service.
- OAuth is used for authorization to access third-party services.

Installation

The Magnet Mobile Server is delivered in binary format. The Mobile Server apk is intended for installation in the system partition of the target mobile device.

System Usage

The Magnet Mobile Server maintains all of its data in the private apps data space. Its SQL database is private and not accessible by other apps. Preferences are also stored in a private file that is not accessible by other apps.

During app installation, the app is temporarily stored on the SD card and removed as soon as the operation completes.

Build Information

- Ubuntu/Linux 12.04 LTS (64 bit)
- Oracle JDK 7
- Maven 3.0.3 or later

Supported Environment

Android SDK

- Java 1.6
- Maven 3.0.3 or later

iOS SDK

- Objective-C language
- Xcode

Documentation

- Magnet Server Application Developer Guide
- Magnet Android Developer Guide
- Magnet iOS Developer Guide
- Magnet Mobile App Server Deployment Guide

Key Features

Release 2.0 includes the following features.

- Caching and off-line message delivery
- Persistence
- Transaction
- Service integration
- Third-party service controllers support

Caching and Off-line Message Delivery

To deal with unreliable connections between mobile devices and the Magnet Mobile Enterprise Server and to ensure reliable message delivery, Magnet Mobile App Server supports off-line operations that store and cache messages, and forwards information when the network connection is reestablished.

Persistence

Magnet Mobile App Server provides support for entities and mapping entities to the data store. This creates, in effect, a "virtual entity data store" over any combination of MySQL, LDAP, and so forth, that can be used from within the programming interface Magnet provides.

Using user-friendly interfaces on entities, the Magnet Mobile App Server provides these features:

- Queries on entity with query composition
- CRUD (Create, Read, Update, Delete) operations on entities
- Complex queries such as paging and ordering

Transactions

The Magnet Mobile App Server can optionally use transactions to maintain the integrity of data.

Service Integration

Service integration can automatically transform Web applications from legacy enterprise application servers to the Magnet Mobile Enterprise Server. It exposes traditional SOAP-based Web Services hosted in the enterprise application servers to REST-based services for mobile access.

Third-Party Service Controllers

The third-party service controllers are sample controllers that allow connection to well-known social services, such as Facebook, LinkedIn, or Salesforce. These controllers have built-in support for OAuth for security and authentication. The source code for these controllers is included in the project generated by the Developer Factory.

Known Issues

ID	Title	Notes
4830	Controller returning mime type application/octet-stream instead of actual mime type	Existing controller returns binary data with mime type application/octet-stream, instead of the actual content type (i.e. image/png or image/jpeg). The client app needs to determine the mime type of the binary data to be able to process it. This issue will be addressed in subsequent releases.
5302	Unsupported javax.xml.datatype.Duration	WSDL using xsd:duration are not supported. Use of xsd:duration is uncommon. This issue will be addressed in subsequent releases.
5303	Unsupported anySimpleType	WSDL using xsd:anySimpleType are not supported. Use of xsd:anySimpleType is uncommon. This issue will be addressed in subsequent releases.
5436	Unexpected UNAUTHENTICATED connection event when using ibinder protocol.	When an Android app connect using the "ibinder" protocol without specifying a username and password, it does not receive AUTHENTICATED connection event although "AUTHENTICATED" event is expected. Instead, it receives UNAUTHENTICATED followed by CONNECTED event. This issue will be addressed in subsequent releases.