

Docomo - Test Strategy 1902

- Performance Test Strategy Review Status
- Performance Testing Strategy
 - Project Overview
 - Project Test Objectives
- Test Methodology
 - Environment Specification
 - Reference Data
 - WS1 UEM proprietary test harness will be used to simulate device and administrator activity in the AirWatch system. Several different scenarios will be executed to determine whether Docomo's architecture will be able to sustain realistic activity for 480,000 devices.
 - Workload Summary
 - Workload A – iOS Device Traffic - Scheduler with APNS and Apple Command Processor
 - Workload B – Android Device Traffic - Scheduler with AWCM and AirWatch Command Processor
 - Workload C – Console UI Test
 - Workload D – Docomo API Usage Test
 - Workload E – Application/Profile/Enrollment Scenarios
- Test Scenarios
 - Test 1:
 - Out of Scope / Won't Test
 - Dependencies
 - Test Result Details
 - Open Issues/Questions/Risks:

Performance Test Strategy Review Status

Reviewer Name	Role	Perf Test Strategy Sent for Review	Perf Test Strategy Reviewed	PE Team incorporated feedback	Notes
Yayoi Yamada	TAM				

Performance Testing Strategy

Project Overview

Docomo has requested to Performance Test all the critical WS1 UEM functionalities with 480K devices.

The goal of this project is to run a variety of tests on an architecture that approximates Docomo's future deployment to gauge the overall performance of WS1 UEM. This information will allow Docomo to understand the Performance of WS1 UEM with 480K devices and help the take critical decisions regarding their future deployment.

Project Test Objectives

- Build environment similar to Docomo's multi-server environment
Please refer this Document for their current deployment - [Docomo_Current_Deployment.pdf](#)
- Model the following with parameters defined by Docomo
 - Device Traffic (iOS and Android)
 - Admin Console Usage
 - API Traffic
 - Application Publish
 - Profile Publish
 - Basic Device Enrollment
- Provide Performance Test results to Docomo and discuss conclusions/recommendations

Test Methodology

Environment Specification

Server	OS	#CPU Cores	Cores per socket	#Sockets	RAM (GB)	Storage	Storage Partitions
1 X Database Server (Partitioned DB)	Windows Server 2012 R2 SQL 2014 - Microsoft SQL Server Enterprise: Core-based Licensing (64-bit)	64	32	2	512	2.38 TB	OS storage (Disk Partition 1) = 80 GB DB storage (Disk Partition 2) = 1.5 TB Log storage (Disk Partition 3) = 200 GB Temp DB storage (Disk Partition 4) = 600 GB

3 X Console Servers	Windows Server 2012 R2	4	2	2	16	80 GB	OS storage (Disk Partition 1) = 80 GB
5 X DS Servers	Windows Server 2012 R2	16	8	2	16	80 GB	OS storage (Disk Partition 1) = 80 GB
3 X API Servers	Windows Server 2012 R2	8	4	2	16	80 GB	OS storage (Disk Partition 1) = 80 GB
5 X AWCN Servers (Implicit Clustering)	Windows Server 2012 R2	4	2	2	8	80 GB	OS storage (Disk Partition 1) = 80 GB
2 x Memcached Server	Red Hat Enterprise Linux 7.3	2	2	1	8	20 GB	OS storage (Disk Partition 1) = 20 GB

Reference Data

WS1 UEM environment specific data			
Airwatch Version		1902	
Cache (Memcache/ AWCACHE)	Memcache	Yes?	
Devices	Android	256,621	Devices
	iOS	213,553	Devices
	Windows	9,826	Devices
Organization Groups	Customer OGs	19,744	OGs
	Container OGs	39,488	
	Device Distribution	Random	
	Largest OG	30,287	Devices
	Smallest OG	1	Devices
Administrators	Total Accounts	59,232	Admins
	Logged In	2,414	Admins/day
Users	Total Accounts	480,000	Users
Smart Groups	All Sizes	277,116	Smart Groups
Applications	Internal	8,465	Apps
	Public	54,703	Apps
	Unmanaged	30,639	Apps
Profiles	All Types	See next slide?	
Content	Documents	N/A	Documents
	Repositories	N/A	Repositories
	Folders	N/A	Folders
Scheduler Settings			
Compliance Policies		38,312	

WS1 UEM proprietary test harness will be used to simulate device and administrator activity in the AirWatch system. Several different scenarios will be executed to determine whether Docomo's architecture will be able to sustain realistic activity for 480,000 devices.

Workload Summary

The test harness is composed of following workloads as described below:

Workload A – iOS Device Traffic - Scheduler with APNS and Apple Command Processor

Overview: iOS traffic load is determined by scheduler.

Details:

- Interrogator samples every 8 hours
- Beacons Samples and command checkin every 1 hour
- Directly proportional to number of devices

- Number of devices specified in Test Scenario

Workload B – Android Device Traffic - Scheduler with AWCM and AirWatch Command Processor

Overview: Android device traffic load is determined by scheduler in Agent app on device.

Details:

- Interrogator samples every 8 hours
- Beacons Samples and command checkin every 1 hour
- Directly proportional to number of devices
 - Number of devices specified in Test Scenario

Workload C – Console UI Test

Overview: Test simulates 35 administrators navigating around the console UI.

Details:

- Number of users – Defined in Test Scenario
- All users will navigate at different OG's, unless otherwise specified
- Ramp up – 1 user every 10 seconds
- Steady state duration – 60 minutes
- Ramp down – 5 users every 30 seconds

Workload D – Docomo API Usage Test

Overview: Test simulates Docomo's anticipated load on the API server.

Considerations:

- Run API calls based off of usage estimates from Docomo
- Below table defines API that will be tested as part of this workload, with the expected rate

No	API	TPS	Duration [sec]
1	Add Console User(Basic)	20	2,500
2	Add Enrollment User(Basic)	20	2,500
3	Register Device for an Enrollment User	20	2,500
4	Search Device Enrollment Token	20	2,500
5	Retrieve Device Information	20	2,500
6	Delete Device	10	2,000
7	Delete List of Enrollment Users	10	2,000

No	API	Method	Version
1	Add Console Admin User(Basic) <code>api/system/admins/addadminuser</code>	POST	Version=1
2	Add Enrollment User(Basic) <code>api/system/users/adduser</code>	POST	Version=1
3	Register Device for an Enrollment User <code>api/system/users/{EnrollmentUserID}/registerdevice</code>	POST	Version=1
4	Search Device Enrollment Token <code>api/system/users/enrollmenttoken/search?organizationgroupid={organizationalgroupid}&serialnumber={serialnumber}</code> Note) Replace replace serialnumber with the serial number of the device and replace organizationgroupid with the OG ID that the device belongs to	GET	Version=1
5	Retrieve Device Information <code>?api/mdm/devices/?searchby={searchby}&id={id}</code> Note) Replace searchby in parenthesis with ImeiNumber and replace ID with the IMEI of the device	GET	Version=1

6	Delete Device ?api/mdm/devices/?searchby={searchby}&id={id} Note) Replace searchby in parenthesis with IimeiNumber and replace ID with the IMEI of the device	DELETE	Version=1
7	Delete List of Enrollment Users api/system/users/delete	POST	Version=1

Workload E – Application/Profile/Enrollment Scenarios

Workload	Scenario
E1	<ol style="list-style-type: none"> 1. Add 10 Managed Internal Applications(4 iOS and 5 Android and 1 Windows) at 10 Customer OGs and and assign it to 480K devices 2. Add 10 Profiles(iOS, Android and Windows) with Passcode, Restrictions, Wifi, Bookmarks and Cert(ADCS CA) and publish it to 480K devices(Cert profile to only 5K) 3. Enrolling 1K devices(500 Android, 450 iOS and 50 Windows) will be running through-out the test.

Test Scenarios

Test 1:

Summary: Simulate WS1 UEM Device load on DS servers

Workloads Tested (Workloads Defined in Methodology Section)

- All workloads are run concurrently
 - Workload A
 - Workload B
 - Workload C
 - Workload D
 - Workload E
 - Workload F

Load Summary (Data Specified in Methodology Section)

- Devices Enrolled – 480,000
- iOS Sample rate – 4 hours
- Android sample rate – 4 hours
- Active Administrators – 35
- Navigation Level – Each individual OG
- API Traffic – X API calls per second (Based on Docomo API Usage)
- Internal Application size - 30 to 40 MB
- Application type - .ipa and .apk
- Smart group size - 20,000 devices
- Each Profile - 1 Payload
- Smart group size - 5,000 devices

Out of Scope / Won't Test

Dependencies

Test Result Details

Open Issues/Questions/Risks: