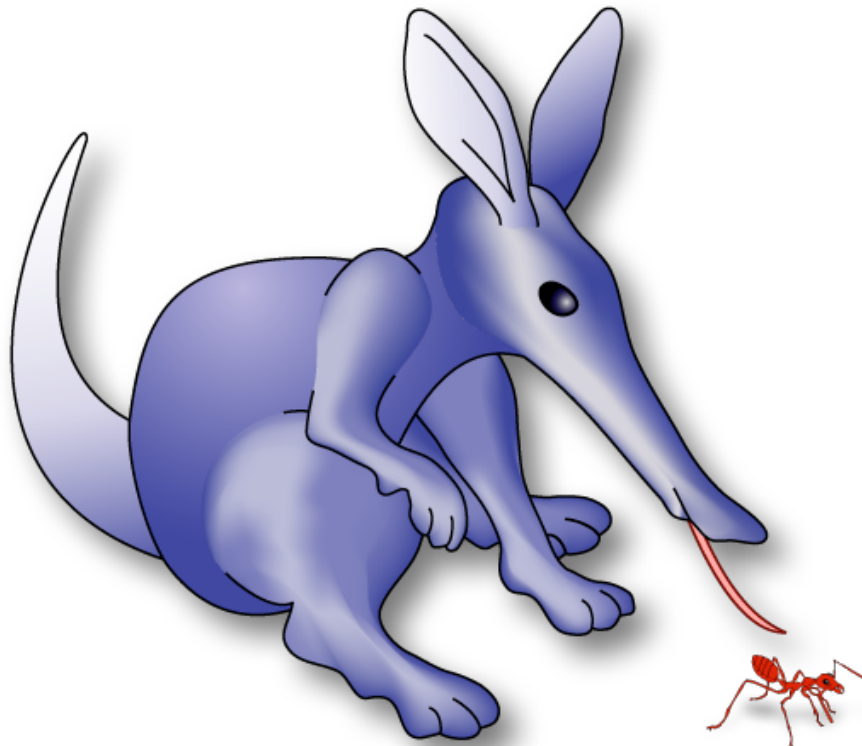


# **Radar Web Services**

## **ENGINEERING REQUIREMENT SPECIFICATION**

**Version 1.6 (0.69)**

Updated : August 21, 2014







## Project Manager

Name	Amul Goswamy
Telephone	+1 408 783 0465
Email	<a href="mailto:amul@apple.com">amul@apple.com</a>

## Project Information

Project Name	Radar Web Services
Project Number	
Project Registration Number	



## Document History

Rev.	Date	Author	Comments
0.1	October 27, 2011	Radar Development Team	Initial Draft
0.2	November 18, 2011	Radar Development Team	Modified operating environment tools and added two more APIs
0.3	November 21, 2011	Radar Development Team	Added 5 more APIs
0.4	November 22, 2011	Radar Development Team	Added 6 more APIs
0.5	November 23, 2011	Radar Development Team	Added 14 more APIs
0.6	November 24, 2011	Radar Development Team	Added 5 more APIs
0.7	November 25, 2011	Radar Development Team	Added 12 more APIs
0.8	November 28, 2011	Radar Development Team	Modifications to Component bundle APIs
0.9	December 05, 2011	Radar Development Team	Added sample HTTP call
0.1	December 08, 2011	Radar Automation Board	Rewrote almost everything
0.11	December 14, 2011	Radar Development Team	Six items completed in Problem Collection APIs
0.12	December 15, 2011	Radar Automation Board	Reorganized and rewrote some problem collection APIs.
0.13	December 16, 2011	Radar Development Team	Reorganized the API Related items in external system and incorporated the review comments.
0.14	December 16, 2011	Radar Automation Board	Cleaned up the problem update and find attributes, and related items in external systems sections.
0.15	December 20, 2011	Radar Development Team	Added third party products, access permissions. keyword and find component APIs, reorganized component APIs



Rev.	Date	Author	Comments
0.16	December 22, 2011	Radar Automation Board	Modified attributes for problem APIs, reorganized keyword APIs.
0.17	December 23, 2011	Radar Development Team	Added component APIs, Person APIs and 5 test script APIs.
0.18	December 30, 2011	Radar Development Team	Added scheduledTest APIs and lookup SQL
0.19	January 9, 2012	Radar Development Team	Added missing APIs
0.2	January 9, 2012	Radar Automation Board	Reviewed component and person APIs.
0.21	January 13, 2012	Radar Development Team	Fixed Review comments
0.22	January 23, 2012	Radar Automation Board	Added section 10, Enumerations, and added details to enclosures, diagnosis, and problem keywords.
0.23	January 25, 2012	Radar Development Team	Added Set Problem Resolver, Rename Problem Enclosures, Get Component Data APIs and Fixed Review comments
0.24	February 7, 2012	Radar Automation Board	Incorporated changes from last meeting, plus feedback from Marc Sinykin.
0.25	February 10, 2012	Radar Automation Board	Added target milestones, changed accessPermissions to securityList, moved scalar fields into main problem attributes list, and other problem attribute changes.
0.26	February 15, 2012	Radar Automation Board	Incorporated changes from last meeting, plus feedback from Marc Sinykin.



Rev.	Date	Author	Comments
0.27	February 20, 2012	Radar Automation Board	Added versioning section, new problem attributes, target milestones changes.
0.28	February 22, 2012	Radar Development Team	Removed ETag header and modified authentication model
0.29	February 28, 2012	Radar Development Team	Added ETag header
0.3	March 9, 2012	Radar Automation Board	Rewrote problem search.
0.31	March 20, 2012	Radar Automation Board	Incorporated feedback from Marc Sinykin.
0.32	March 22, 2012	Radar Automation Board	Incorporated more feedback from Marc, and cleaned up the keyword and component sections.
0.33	April 3, 2012	Radar Development Team	Added Find Component Request and response parameters.
0.34	April 6, 2012	Radar Automation Board	Rewrote some of the Find Component section, added responses for all update/add APIs, and fixed a few other attributes.
0.35	April 6, 2012	Radar Automation Board	Replaced ETag with fingerprint attribute.
0.36	April 17, 2012	Ewalt	clearly indicate what's required to set originator on new problem.
0.37	April 20, 2012	Radar Automation Board	Removed server responses for update/append actions.
0.38	April 24, 2012	Radar Development Team	Updated Request parameter for Find Problem and incorporated reviews on TestSuite APIs
0.39	May 22, 2012	Radar Automation Board	Rewrote Find Problem section again.



Rev.	Date	Author	Comments
0.4	June 1, 2012	Radar Automation Board	Added problem history section, and changed recordHistory search key to everAssignedTo.
0.41	June 12, 2012	Radar Development Team	Added details on authentication models supported and examples
0.42	July 23, 2012	Radar Development Team	Added the URLs to access the web-services on Prod / UAT
0.43	July 26, 2012	Radar Development Team	Modified personnel field of FindComponent.
0.44	Aug 04, 2012	Radar Development Team	Added extra fields on Problem History Response
0.45	Aug13, 2012	Radar Development Team	Added keywords, testCycle in FindScheduleTest response and Added Response object for Create TestSuite and AddScheduledTest API
0.46	Aug 16, 2012	Radar Development Team	Modified example of Add case to scheduled test and Set schedule test case data.
0.47	Aug 30, 2012	Radar Development Team	Modified Response parameters for FindProblem API
0.48	Sep 19, 2012	Radar Development Team	Included keywordNames attribute in GetProblemByID
0.49	Oct 10, 2012	Radar Development Team	Corrected the mismatch examples.
0.5	Oct 26, 2012	Radar Development Team	Merge the ERS of WebServices 1.1 in 1.0
0.51	Jan 14, 2013	Radar Development Team	< <a href="#">rdar://problem/12688397&amp;12772823&amp;12772844&amp;12772861&amp;12813514&amp;12857797&amp;12886562&amp;12893710</a> >



Rev.	Date	Author	Comments
0.53	Feb 1, 2013	Hackamack	<a href="rdar://problem/13105988&amp;13106006&amp;13106098&amp;13106135&amp;13106146&amp;13106335&amp;13106481&amp;13107931">rdar://problem/13105988&amp;13106006&amp;13106098&amp;13106135&amp;13106146&amp;13106335&amp;13106481&amp;13107931</a>
0.54	Feb 7, 2013	Hackamack	Formatting changes to make the TOC easier to read and standardizing on table formats. <a href="rdar://problem/13106015&amp;13105924">rdar://problem/13106015&amp;13105924</a>
0.55	Feb 11, 2013	Radar Development Team	Reflecting the url changes - <a href="rdar://problem/12600994">rdar://problem/12600994</a> Remove service prefix from radar web service url
0.56	Feb 14, 2013	Radar Development Team	13190137,13054065,12417452,13095264 ,13188342
0.57	Mar 8, 2013	Radar Development Team	Merging all WS 1.2 Changes
0.58	June 24, 2013	Radar Development Team	<a href="rdar://problem/13252256&amp;13563749&amp;13563752&amp;13563758&amp;13563761">rdar://problem/13252256&amp;13563749&amp;13563752&amp;13563758&amp;13563761</a>
0.59	July 18, 2013	Radar Development Team	<a href="exp2://Ticket/15798252">exp2://Ticket/15798252</a> RWS : Bug fixes on 19 july 2013
0.6	Sep 20, 2013	Radar Development Team	<a href="rdar://problem/14763273&amp;14855176&amp;14891622&amp;14996748">rdar://problem/14763273&amp;14855176&amp;14891622&amp;14996748</a>
0.61	Oct 25, 2013	Radar Development Team	<a href="rdar://problem/12342955&amp;15142779">rdar://problem/12342955&amp;15142779</a>
0.62	Feb 05, 2014	Radar Development Team	Merging all WS 1.4 Changes
0.63	Apr 15, 2014	Radar Development Team	Merging all RWS-1.5 Changes
0.64	May 20, 2014	Radar Development Team	Merging all RWS-1.5.1 Changes
0.65	May 21, 2014	Amul Goswamy	Updated auth recommendation
0.66	July 06, 2014	Radar Development Team	Merging all RWS-1.5.2 Changes
0.67	August 20, 2014	Radar Development Team	Reflecting Bug fixes for 1.6
0.68	August 21, 2014	Radar Development Team	Reflecting API changes for 1.6





Rev.	Date	Author	Comments
0.69	August 21, 2014	Amul Goswamy	Final revision for 1.6 release.



## ***Table of Contents***



<b>1. INTRODUCTION.....</b>	<b>18</b>
<b>1.1 Document Overview.....</b>	<b>18</b>
<b>1.2 Scope of the Project .....</b>	<b>18</b>
<b>1.3 Assumption.....</b>	<b>18</b>
<b>2. OPERATING ENVIRONMENT .....</b>	<b>19</b>
<b>2.1 Tools.....</b>	<b>19</b>
<b>2.2 Network &amp; Communication .....</b>	<b>19</b>
<b>2.3 Connection URLs .....</b>	<b>19</b>
<b>2.4 Authentication Model.....</b>	<b>19</b>
2.4.1 SSO AUTHENTICATION SCHEME .....	19
2.4.2 SPNEGO AUTHENTICATION .....	20
2.4.3 APPLE CONNECT MOBILE ( ACM ) TOKEN BASED AUTHENTICATION.....	21
2.4.4 OPAQUETOKEN BASED AUTHENTICATION .....	22
<b>2.5 Error Messages and Status.....</b>	<b>22</b>
2.5.1 ERROR MESSAGE FORMAT.....	22
2.5.2 STATUS CODES .....	23
<b>2.6 API Versioning .....</b>	<b>24</b>
<b>3. PROBLEMS.....</b>	<b>25</b>
<b>3.1 Get Problem By ID.....</b>	<b>25</b>
<b>3.2 Find Problems .....</b>	<b>47</b>
<b>3.3 Create New Problem .....</b>	<b>81</b>
<b>3.4 Update Problem.....</b>	<b>86</b>
<b>3.5 Modify Multiple Problems.....</b>	<b>94</b>
<b>3.6 Clone Problem.....</b>	<b>102</b>
<b>3.7 Create New Problem With Attachments and Pictures .....</b>	<b>104</b>
<b>3.8 Get Problem Protection Mask .....</b>	<b>112</b>
<b>3.9 Problems Included in Other Objects .....</b>	<b>113</b>
<b>4. PROBLEM COLLECTIONS .....</b>	<b>115</b>
<b>4.1 Problem Description .....</b>	<b>115</b>



4.1.1 GET PROBLEM DESCRIPTION .....	115
4.1.2 APPEND TO PROBLEM DESCRIPTION .....	117
<b>4.2 Problem Diagnosis.....</b>	<b>117</b>
4.2.1 GET PROBLEM DIAGNOSIS.....	118
4.2.2 APPEND TO PROBLEM DIAGNOSIS.....	119
<b>4.3 Problem Enclosures .....</b>	<b>120</b>
4.3.1 GET PROBLEM ENCLOSURES LIST .....	121
4.3.2 DOWNLOAD PROBLEM ENCLOSURE.....	126
4.3.3 UPLOAD PROBLEM ENCLOSURE.....	128
4.3.4 DELETE PROBLEM ENCLOSURE.....	130
4.3.5 MODIFY PROBLEM ENCLOSURE.....	130
4.3.6 GET PROBLEM PICTURE THUMBNAIL .....	132
<b>4.4 Related Problems.....</b>	<b>132</b>
4.4.1 GET RELATED PROBLEMS LIST .....	132
4.4.2 SET RELATED PROBLEMS LIST .....	134
4.4.3 ADD RELATED PROBLEM.....	135
4.4.4 REMOVE RELATED PROBLEM .....	136
4.4.5 EDIT RELATED PROBLEM RELATION .....	137
4.4.6 FIND RELATED PROBLEMS .....	140
<b>4.5 Problem Keywords.....</b>	<b>143</b>
4.5.1 GET PROBLEM KEYWORDS LIST .....	143
4.5.2 SET PROBLEM KEYWORDS LIST.....	146
4.5.3 ADD KEYWORD TO PROBLEM.....	147
4.5.4 REMOVE PROBLEM KEYWORD.....	148
4.5.5 GET PROBLEM KEYWORD.....	149
4.5.6 ADD MULTIPLE KEYWORDS TO ACTIVE/INACTIVE PROBLEM .....	150
4.5.7 REMOVE MULTIPLE KEYWORDS FROM ACTIVE/INACTIVE PROBLEM .....	151
<b>4.6 Other Related Items .....</b>	<b>153</b>
4.6.1 GET OTHER RELATED ITEMS LIST.....	153
4.6.2 GET OTHER RELATED ITEMS SYSTEMS LIST.....	155
4.6.3 SET LIST OF OTHER RELATED ITEMS .....	156
4.6.4 ADD OTHER RELATED ITEM.....	157
4.6.5 EDIT OTHER RELATED ITEM.....	158
4.6.6 REMOVE OTHER RELATED ITEM .....	159
<b>4.7 Third Party Products.....</b>	<b>159</b>
4.7.1 GET THIRD PARTY PRODUCT LIST.....	160
4.7.2 SET THIRD PARTY PRODUCT LIST .....	161
4.7.3 ADD THIRD PARTY PRODUCT .....	162
4.7.4 REMOVE THIRD PARTY PRODUCT .....	163
4.7.5 FIND THIRD PARTY PRODUCT .....	163
<b>4.8 Security List.....</b>	<b>165</b>
4.8.1 GET SECURITY LIST .....	165
4.8.2 ADD PERSON OR GROUP TO SECURITY LIST .....	167
4.8.3 REMOVE PERSON OR GROUP FROM SECURITY LIST .....	168



4.8.4 APPEND TO SECURITY LIST.....	169
4.8.5 GET PROBLEM PRIVILEGES.....	170
<b>4.9 CC List.....</b>	<b>171</b>
4.9.1 GET CC LIST.....	171
4.9.2 ADD PERSON TO CC LIST .....	172
4.9.3 REMOVE PERSON FROM CC LIST .....	173
<b>4.10 Target Milestones .....</b>	<b>173</b>
4.10.1 GET TARGET MILESTONES LIST .....	174
4.10.2 ADD TARGET MILESTONE TO PROBLEM .....	175
4.10.3 REMOVE TARGET MILESTONE FROM PROBLEM .....	177
4.10.4 CLONE PROBLEM TO TARGET MILESTONES .....	178
4.10.5 EDIT TARGET MILESTONES PLACEHOLDERS.....	180
<b>4.11 Problem History.....</b>	<b>181</b>
4.11.1 GET PROBLEM HISTORY .....	181
4.11.2 GET PROBLEM DIAGNOSIS HISTORY.....	184
<b>4.12 Problem Crash APIs.....</b>	<b>193</b>
4.12.1 UPDATE PROBLEMS CRASH COUNT .....	193
4.12.2 GET CRASH TRACER RADAR LIST .....	194
4.12.3 GET CRASH DUPLICATE CHAIN .....	195
<b>4.13 Product Security Targets.....</b>	<b>197</b>
4.13.1 GET TARGETS FOR PROBLEM .....	197
4.13.2 ADD TARGET TO PROBLEM .....	199
4.13.3 EDIT TARGET OF PROBLEM.....	200
4.13.4 REMOVE TARGET OF PROBLEM .....	201
<b>4.14 Product Security Reporters .....</b>	<b>202</b>
4.14.1 GET REPORTERS FOR PROBLEM .....	202
4.14.2 ADD REPORTER TO PROBLEM .....	203
4.14.3 EDIT REPORTER OF PROBLEM.....	204
4.14.4 REMOVE REPORTER OF PROBLEM .....	205
<b>4.15 Product Security ExternalIDs.....</b>	<b>206</b>
4.15.1 GET EXTERNALIDS FOR PROBLEM .....	206
4.15.2 ADD EXTERNALID TO PROBLEM .....	207
4.15.3 EDIT EXTERNALID OF PROBLEM.....	207
4.15.4 REMOVE EXTERNALID OF PROBLEM .....	208
<b>4.16 Product Security Product.....</b>	<b>209</b>
4.16.1 GET PRODUCTS LIST .....	209
4.16.2 CREATE PRODUCT.....	210
4.16.3 OPEN PRODUCT .....	212
4.16.4 UPDATE PRODUCT .....	213
4.16.5 DELETE PRODUCT .....	213
4.16.6 REMOVE GROUP FROM PRODUCT.....	214
<b>4.17 Product Security Release Vehicle .....</b>	<b>215</b>



4.17.1 GET RELEASE VEHICLES LIST .....	215
4.17.2 CREATE RELEASE VEHICLE.....	216
4.17.3 OPEN RELEASE VEHICLE .....	217
4.17.4 UPDATE RELEASE VEHICLE .....	218
4.17.5 DELETE RELEASE VEHICLE .....	219
4.17.6 REMOVE GROUP FROM RELEASE VEHICLE.....	220
<b>4.18 SSP (SOAP/XML) based services .....</b>	<b>220</b>
4.18.1 VALIDATE RADAR ID .....	220
4.18.2 UPDATE RADAR PROBLEM.....	221
4.18.3 GET SCHEDULED TEST .....	222
4.18.4 FETCH PROBLEM DETAILS .....	222
4.18.5 RELATED SONAR .....	222
4.18.6 GET PROBLEM STATISTICS .....	223
<b>4.19 Related Tests .....</b>	<b>228</b>
4.19.1 GET RELATED TESTS.....	229
4.19.2 EDIT RELATED TESTS .....	230
4.19.3 ADD RELATED TESTS.....	231
4.19.4 REMOVE RELATED TESTS .....	232
<b>4.20 Watched Problems .....</b>	<b>232</b>
4.20.1 ADD PROBLEM TO WATCHED PROBLEM LIST.....	232
4.20.2 REMOVE PROBLEM FROM WATCHED PROBLEM LIST.....	233
<b>4.21 Get Configuration Text.....</b>	<b>234</b>
<b>4.22 Query APIs .....</b>	<b>235</b>
4.22.1 GET RECENT QUERY LIST .....	235
4.22.2 GET INDIVIDUAL RECENTLY EXECUTED QUERY .....	237
4.22.3 EXECUTE RECENT QUERY API.....	238
4.22.4 EXECUTE SHARED REPORT .....	240
4.22.5 GET SHARED REPORTS.....	245
4.22.6 CREATE SHARED REPORT.....	246
4.22.7 GET SHARED REPORT SUBSCRIBER LIST .....	249
4.22.8 GET QUERY DETAILS API .....	250
4.22.9 GET QUERY SUBSCRIBED API.....	253
4.22.10 ADD AND EDIT SUBSCRIBERS IN SHARED REPORT .....	255
4.22.11 REMOVE SUBSCRIBERS FROM SHARED REPORT.....	258
4.22.12 GET RECENTLY OPENED PROBLEMS .....	259
4.22.13 GET RECENTLY OPENED TEST SUITES.....	261
4.22.14 GET RECENTLY OPENED TEST SUITE CASES.....	264
4.22.15 GET RECENTLY OPENED SCHEDULED TESTS.....	266
4.22.16 GET RECENTLY OPENED SCHEDULED TEST CASES .....	269
<b>4.23 Draft APIs .....</b>	<b>271</b>
4.23.1 CREATE PROBLEM DRAFT .....	271
4.23.2 DELETE PROBLEM DRAFT.....	278
4.23.3 GET ALL DRAFTS .....	279
4.23.4 GET PROBLEM DRAFT BY ID.....	280
4.23.5 UPDATE PROBLEM DRAFT .....	284



4.23.6 GET DRAFTS USAGE.....	291
<b>5. KEYWORDS.....</b>	<b>293</b>
<b>5.1 Get Keyword by ID .....</b>	<b>293</b>
<b>5.2 Find Keywords .....</b>	<b>294</b>
<b>5.3 Keywords Included in Other Objects .....</b>	<b>296</b>
<b>6. COMPONENTS.....</b>	<b>297</b>
<b>6.1 Get Component by Name and Version.....</b>	<b>297</b>
<b>6.2 Find Components.....</b>	<b>300</b>
<b>6.3 Component Builds, Milestones and Event .....</b>	<b>305</b>
6.3.1 ADD BUILD OR EVENT TO COMPONENT .....	305
6.3.2 GET COMPONENT BUILDS OR EVENTS OR MILESTONE .....	307
6.3.3 ADD MILESTONE TO COMPONENT .....	309
6.3.4 ADD ACCESS GROUP TO MILESTONE .....	310
6.3.5 REMOVE ACCESS GROUP FROM MILESTONE .....	311
6.3.6 EDIT COMPONENT EVENT, MILESTONE , OR BUILD.....	312
6.3.7 REMOVE COMPONENT EVENT, MILESTONE , OR BUILD .....	315
6.3.8 COMPONENT BUILDS AND MILESTONES INCLUDED IN OTHER OBJECTS .....	316
<b>6.4 Component Bundles .....</b>	<b>317</b>
6.4.1 CREATE COMPONENT BUNDLE .....	317
6.4.2 ADD COMPONENT TO BUNDLE .....	318
6.4.3 REMOVE COMPONENT FROM BUNDLE .....	319
6.4.4 GET COMPONENT BUNDLE.....	320
6.4.5 MODIFY COMPONENT BUNDLE.....	328
6.4.6 GET ALL COMPONENT BUNDLE LIST.....	329
<b>6.5 Fetch component tree.....</b>	<b>330</b>
<b>6.6 Fetch component root details .....</b>	<b>335</b>
<b>6.7 Components Included in Other Objects .....</b>	<b>337</b>
<b>7. PEOPLE .....</b>	<b>338</b>
<b>7.1 Find People.....</b>	<b>338</b>
<b>7.2 People Included in Other Objects.....</b>	<b>340</b>
<b>7.3 Fetching details of logged in person .....</b>	<b>341</b>
<b>8. TEST SUITES AND SCHEDULED TESTS .....</b>	<b>343</b>
<b>8.1 TestSuite .....</b>	<b>343</b>
8.1.1 CREATE TESTSUITE.....	343



8.1.2 ADD CASE TO TESTSUITE .....	345
8.1.3 SET TESTSUITE DATA .....	347
8.1.4 SET TESTSUITE CASE DATA .....	348
8.1.5 GET TESTSUITE DATA .....	351
8.1.6 REMOVE TESTSUITE CASE .....	357
8.1.7 FIND TEST SUITE .....	357
8.1.8 GET TEST SUITE ENCLOSURES LIST .....	364
8.1.9 DOWNLOAD TEST SUITE ENCLOSURE .....	370
8.1.10 UPLOAD TEST SUITE ENCLOSURE .....	371
8.1.11 DELETE TEST SUITE ENCLOSURE .....	373
8.1.12 MODIFY TEST SUITE ENCLOSURE .....	374
<b>8.2 Scheduled Test .....</b>	<b>375</b>
8.2.1 ADD SCHEDULED TEST .....	375
8.2.2 ADD CASE TO SCHEDULED TEST .....	377
8.2.3 REMOVE SCHEDULED TEST CASE .....	379
8.2.4 SET SCHEDULED TEST CASE DATA .....	380
8.2.5 FIND SCHEDULED TEST .....	383
8.2.6 GET SCHEDULED TEST DATA .....	391
8.2.7 SET SCHEDULED TEST DATA .....	399
8.2.8 GET SCHEDULED TEST ENCLOSURES LIST .....	400
8.2.9 DOWNLOAD SCHEDULED TEST ENCLOSURE .....	406
8.2.10 UPLOAD SCHEDULED TEST ENCLOSURE .....	408
8.2.11 DELETE SCHEDULED TEST ENCLOSURE .....	409
8.2.12 MODIFY SCHEDULED TEST ENCLOSURE .....	410
<b>9. RAW SQL EXECUTION .....</b>	<b>412</b>
9.1 LookUp SQL .....	412
<b>10. ENUMERATIONS .....</b>	<b>414</b>
10.1 GET FIELD ENUMERATION .....	414
<b>11. LABELS COLLECTION .....</b>	<b>433</b>
11.1 Get Labels .....	433
11.2 Label Set Labels .....	434
11.2.1 GET LABELS FROM LABEL SET .....	435
11.2.2 ADD LABEL TO LABEL SET .....	436
11.2.3 EDIT LABEL IN LABEL SET .....	437
11.2.4 DELETE LABEL FROM LABEL SET .....	438
11.3 Find Label Sets .....	438
11.4 Subscribe Label Set .....	440
11.4.1 GET SUBSCRIBED LABEL SETS .....	440
11.4.2 SUBSCRIBE A LABEL SET .....	441
11.4.3 UNSUBSCRIBE A LABEL SET .....	442
11.5 Active Label Set .....	442





11.5.1 GET ACTIVE LABEL SET .....	442
11.5.2 SET ACTIVE LABEL SET .....	444
<b>12. FAVORITE PREFERENCE APIS .....</b>	<b>445</b>
12.1 GET FAVOURITE SHARED REPORT .....	445
12.2 ADD FAVOURITE FOR SHARED REPORTS .....	448
12.3 REMOVE FAVOURITE FOR SHARED REPORTS .....	448
12.4 GET FAVOURITE PEOPLE .....	449
12.5 ADD TO FAVOURITE FOR PEOPLE .....	450
12.6 REMOVE FROM FAVOURITE FOR PEOPLE .....	450
12.7 GET FAVOURITE COMPONENT .....	451
12.8 ADD TO FAVOURITE FOR COMPONENTS .....	452
12.9 REMOVE FROM FAVOURITE FOR COMPONENTS .....	452
12.10 GET FAVOURITE OTHER RELATED ITEMS .....	453
12.11 ADD FAVOURITE FOR OTHER RELATED ITEMS .....	454
12.12 REMOVE FAVOURITE FOR OTHER RELATED ITEMS .....	454
12.13 GET FAVOURITE LAYOUTS .....	455
12.14 ADD LAYOUTS TO FAVORITES .....	457
12.15 REMOVE FAVORITE FOR LAYOUTS .....	457
<b>13. GROUP .....</b>	<b>459</b>
13.1 GET WORK-GROUP DETAILS .....	459
13.2 GET ACCESS-GROUP DETAILS .....	463
13.3 CREATE WORK-GROUP .....	466
13.4 CREATE ACCESS-GROUP .....	467
13.5 UPDATE WORK-GROUP .....	469
13.6 UPDATE ACCESS-GROUP .....	470
13.7 ADD PERSONS TO MEMBERS .....	471
13.8 SET PRIVILEGE FOR THE MEMBER .....	473
13.9 REMOVE GROUP MEMBERS .....	474
13.10 ADD ADMINISTRATOR .....	474
13.11 REMOVE ADMINISTRATOR .....	475
13.12 FIND GROUP .....	475
<b>14. SIGNING ON AND OFF .....</b>	<b>479</b>
14.1 SIGN-ON TO RADAR WITH SINGLE-SIGN-ON (SSO) CREDENTIALS .....	479
14.2 SIGN OFF FROM RADAR API .....	480



## 1. INTRODUCTION

### 1.1 Document Overview

This document outlines the requirements for converting superset of Radar AppleScript and CLI APIs to web services. Effort will be made to make them generic. The document will list:

- Description of each web services.
- Business logic to be used to address the requirement/problem.

The design and implementation of the system will be based upon the information specified in this document.

### 1.2 Scope of the Project

The scope of this project is to design, construct and implement a set of Radar Web Services. The exact details of these web services are mentioned in the rest of the document that will together clearly define the above-mentioned high-level scope of the project.

#### [A] Inclusion

- Development of Web Services.
- Development of Web service Client (for Testing and publishing of new web services)

#### [B] Exclusion

- Any item not specifically mentioned in section 'Inclusions'

### 1.3 Assumption

- Integration Testing will happen via Test client developed to test new APIs



## 2. OPERATING ENVIRONMENT

### 2.1 Tools

Client	HTTP Protocol
Hilo Server	Version 3.0.5 or later
IDE	STS 2.8.1
Web-Service	Spring 3 MVC

### 2.2 Network & Communication

Radar currently supports TCP/IP. There will be no change to it in Radar Web Service.

### 2.3 Connection URLs

Radar Web-services can be accessed through the following URLs:

Production:

<https://radar-webservices.apple.com>

UAT:

<https://bugreport-test-new.apple.com>

### 2.4 Authentication Model

The web-services have the following authentication methods supported

#### 2.4.1 SSO Authentication Scheme

##### 2.4.1.1 Description

Radar API implements an OAuth 2.0-flavored authentication scheme. On behalf of the user, clients obtain Single-Sign-On (SSO) credentials from Apple's Identity Management Service (IdMS). SSO credentials are needed in order to obtain Radar Authentication credentials. Radar Authentication credentials included as an HTTPS Header of an API call will be honored until the session expires. The Radar Authentication credentials is as simple as setting the access token as the value for a header named "Radar-Authentication" as demonstrated in this example verbose output from a cURL command:

```
GET /problems/4000000 HTTP/1.1
User-Agent: curl/7.30.0
Host: bugreport-test-new.apple.com
```



```
Accept: application/json
Radar-Authentication:kjfkjaije;33902uj;js0033jksjij3
```

Initially, a call to any Radar API call will accept either SSO credentials or Radar Authentication credentials. A change is planned to only accept SSO credentials at the */signon* endpoint in order to get Radar Authentication credentials. In the future, an announcement will be circulated instructing client systems to make the necessary changes and will specify the date of release for this change. Offering Radar Authentication as optional for a period of time is to ensure a smooth transition for all Radar API users.

#### 2.4.1.1 Recommended Usage

The Radar Team recommends native clients and server-to-server clients adopt Radar Authentication according to the flow in this outline:

1. Obtain SSO credentials (sign-on to IdMS if needed).
2. Sign-on to Radar using the */signon* endpoint. Refer Sec 14.1 SSO Login RWS for Details
3. Retain the *access\_token* and include Radar-Authentication header in all HTTPS Requests.
4. Retain the *expires\_in* number of seconds.
5. At the time the session expires, perform steps 1 through 4.
6. Call the */signoff* endpoint to close the session when activity is complete (if applicable; e.g. the native application quitting, user signing out of IdMS, a batch job completes successfully or some unexpected security event. Refer Sec 14.2 SSO Logout RWS for Details

#### 2.4.2 SPNego Authentication

This is the most generic method of authentication, and uses the kerberos token generated when the user logs-in into Apple Connect. The kerberos token is sent to the server, which then validates it against DS using GSS APIs, and appropriately provides the successful results or the authentication denied error response.

It works as follows:

1. The user logs in into Apple Connect using kinit or other processes.
2. Use Curl to request for the resource from App Server (make the web-service call)
3. The server sees that there is no valid session associated with request and it is not authenticated so far. So, it sends out a response indicating "WWW-Authenticate : Negotiate" header back to the client.
4. The client will see the 401 Unauthorized status and since the Authenticate header indicates "Negotiate", it will use the current security context created by "kinit" and uses the details of the service "protocol/hostname@domainname" to get the service token needed for authentication.
5. The client will then send this service token in the "Authorization: Negotiate " header to the server, which will be Base-64 encoded.
6. The server will use this token, authenticate against the kerberos distribution center in which is Appleconnect (DS). Once the authentication is successful, the requested resource is sent back to the



client with HTTP 200 OK status and in case of error, the response is sent back to the client in the WWW-Authenticate header in Base 64 encoding.

#### 2.4.2.1 SPNego known issues

Issues with libcurl on certain versions of 10.7.x prevent SPNego Authentication from properly sending along the Kerberos service token to the App Server. For details, see <[rdar://problem/13095264](http://rdar://problem/13095264)> Radar Web Services: SPNego authentication fails for 10.7 and 10.6 clients

To identify the version execute the below command

```
strings /usr/lib/libcurl.4.dylib | grep curl | grep SourceCache
/SourceCache/curl/curl-68/curl/lib/ssluse.c
```

More details on the issue and the fix can be found under the tracking radar <[rdar://problem/10292881](http://rdar://problem/10292881)> Duchess: CVE-2011-2192: libcurl inappropriate GSSAPI delegation 2.4.1 DS Cookie based Authentication

Though this method is commonly used by RWS clients, WebAuth-derived Cookie-based authentication is the least preferred method of authentication. This cookie is intended for authenticating browser-based web applications only. A DS Cookie (DS Web Auth Token) sent to the server as part of the request in the web-service call is still used by the server to support projects that have not yet migrated their systems. Until a company-wide migration date is set, the server validates this cookie against Directory Services for the respective user and accordingly provides the successful output or authentication denied error message.

The user can generate a cookie by using either the cookie generator script included in the Radar Web Services examples from <http://radar.apple.com>.. The cookie generated will be of the name myacinfo (or myacinfo-uat for test environment), which is a 128-bit encrypted string.

#### 2.4.3 Apple Connect Mobile ( ACM ) Token based Authentication

This method of authentication is supported for iOS client like iPad or iPhone. In this method of authentication, the ACM Token (AppleConnectMobile Token) is sent to the server as part of the request in the web-service call. The server validates this token against Directory Services for the respective user and accordingly provides the successful output or authentication denied error message.

The user can generate the token by using Xcode build present at <http://max.apple.com/Releases/Production/Current/ACMobileInternal.tgz> which will give a token after entering AppleConnectUserName and Password. This token needs to be URL encoded first before sending to Web Service server. After URL encoding the token, it needs to be send to server as a cookie in the request as mentioned below.

Client Request:-

```
curl -i -H "Accept: application/json" --cookie
"WEB_COOKIE_ENV=ACMTKNeNpdV...ve%2Bz" -H "Content-Type: application/
json" https://bugreport-test-new.apple.com/problems/10187518
```

```
For production
WEB_COOKIE_ENV = myacinfo
```

```
For UAT
WEB_COOKIE_ENV = myacinfo-uat
```



Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Set-Cookie: JSESSIONID=I0arRKxlTl0dqX3JuCloUrnH.node36656; Path=/;
Secure
Content-Type: application/json; charset=UTF-8
{
  "assignee": {
    "dsid": 17856935,
    "email": "adc_bugs@apple.com",
    "firstName": "Developer",
    "lastName": "ADC Bugs",
    "type": "Contractor"
  },
  "assigneeLastModifiedAt": "2012-06-04T17:05:00+0000",
  "attachmentsCount": 0,
  "ccCount": 0,
  ...more records
}
```

#### 2.4.4 OpaqueToken based Authentication

Authentication with the Opaque Token is not permitted by IS&T Information Security. Radar is in the process of phasing out their use. Please see <[rdar://problem/16457974](http://rdar://problem/16457974)> for more details.

## 2.5 Error Messages and Status

Error messages should follow a common JSON output format (defined below) and should use an appropriate HTTP status code.

### 2.5.1 Error Message Format

Error messages should be encoded as a JSON object with the following format:

```
{
  "help": "View documentation at http://radar.apple.com/",
  "message": "The problem you are trying to modify has already been
modified by another user. Please reload the problem before saving
again.",
  "status": "409 Conflict",
  "title": "Problem has been modified"
}
```

The status property should contain the HTTP status line.

The title property should contain a one-line title that explains why this error occurred.

The message property should contain a more thorough explanation as to why the error occurred, and if possible, steps they can take to prevent the error from happening again.



The help property should contain information about how the user can gain more information to help them resolve the error. This should be (for example) a wiki or an email address.

The optional errors property contains an array of error details. Each error will contain “title” and “message” fields, as well as an optional “attribute” field. The “attribute” field indicates which attribute was in error. If the error was not related to a particular attribute, the field may be omitted.

E.g., in the case of a problem update with an ambiguously specified keyword:

```
{
  "errors": [
    {
      "attribute": "keyword",
      "message": "The keyword name \u2018foo\u2019 is not unique.
Search for a keyword and use the id instead.",
      "title": "name not unique"
    }
  ],
  "help": "View documentation at http://radar.apple.com/",
  "message": "The problem couldn\u2019t be updated due to a conflict
in the keyword field.",
  "status": "409 Conflict",
  "title": "Unable to update problem due to conflicts"
}
```

## 2.5.2 Status Codes

The following HTTP status codes should be used in the following scenarios:

- **200 OK:** Standard response for successful HTTP requests.
- **201 Created:** The request has been fulfilled and resulted in a new resource being created.
- **204 No Content:** The server successfully processed the request, but is not returning any content.
- **400 Bad Request:** This should be used in POST or PUT requests when the data does not permit the action to complete successfully. Examples of this may include missing required fields. Another example might be if the user specified a state of "Integrate" along with a substate.
- **401 Not Authorized:** This should be used when the request does not include any valid authentication credentials.
- **403 Forbidden:** This should be used when the user has successfully authenticated but does not have permission to perform the action.
- **404 Not Found:** This should be used whenever the user tries to access a resource that does not exist.
- **408 Request Timeout:** This should be used when the user initiates a request that takes longer than 5 minutes to complete, such as an overly broad problem search.
- **409 Conflict:** This should be used in POST or PUT requests where the data conflicts with some condition on the server. An example of this is if the problem has been modified from the version the user is trying to save.
- **500 Internal Server Error:** This should be used when any exception happens at Application server end and it is unable to process the WS request.
- **502 Bad Gateway:** If the web services failed to response within web server time limit then web server will drop the connection and this status will be returned in response.



- **503 Service Unavailable:** If the web services are unable to perform the request due to a backend availability issue, this status code should be used.

For more information on HTTP status codes, please consult the RFC ( <http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html> ).

## 2.6 API Versioning

In order to avoid breaking clients who depend on old versions of APIs, a versioning scheme is provided via the `X-API-Version` header. The `X-API-Version` header specifies which version of the API to call against. Any client that doesn't specify the API version will receive the latest version of the API. The server response uses the `X-API-Version` header to indicate the version that it is using.

Currently, the server will always respond with `application/json` (except in the case of attachments), regardless of the client's accepted formats.

### [A] Header Format

```
X-API-Version: <version>
```

### [B] Examples

#### Request a problem with version 1.3

Client request:

```
GET /problems/10000000
X-API-Version: 1.3
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.3
```

...





### 3. PROBLEMS

These APIs provide fetching, searching, creation, and modification of Radar problem records. In general, collections (such as relatedProblems) can also be fetched or modified along with a problem. See [3.1 Get Problem By ID](#) for Description of the formats of these collections.

#### 3.1 Get Problem By ID

##### [A] Description

This API is used to get data from a problem or from multiple problems by specifying the problem ID or IDs in the url. Fields, including scalar columns and collections, can be specified using the X-Fields-Requested header.

On a successful response, the HTTP status is 200 OK. When an error occurs, an appropriate HTTP response code is used, along with a JSON representation of the error.

A 'fingerprint' attribute will be return with each problem. It is a hash encoded value of lastModifiedAt. The fingerprint attribute can be used to check whether any field in problem is changed or not, as when any problem attribute changes, its lastModifiedAt date will be also changed, in which case fingerprint will also be changed to new value.

##### [B] Schedule

Required for milestone 1.0

##### [C] URL Scheme

```
GET /problems/<problem_id>[,<problem_id>]
```

##### [D] Response Attributes

The default set of attributes (indicated in the rightmost column) can be overridden with the X-Fields-Requested header. Any attributes that have no value will be returned with a value of null: if the database contains an empty string for Data Type that allow null, the service will return the value null instead of "".

Key	Description	Data Type	Default
id	Unique ID of the problem	Integer	Y
fingerprint	A string that the server uses to validate that the problem is the most recent version. It is a Hash Encoded value of lastModifiedAt.	String	Y
title	Title of the problem. Maximum size of 240 Characters.	String	Y



Key	Description	Data Type	Default
component	The component that the problem belongs to. (See <a href="#">6.5 Components Included in Other Objects</a> )	Object	Y
originator	The person who created the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Object	Y
assignee	The person who is assigned to the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Object	Y
dri	The person who is the DRI for the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Object or null	N
proxy	The person who is designated as the proxy for this problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Object or null	N
epm	The person who is the EPM for the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Object or null	N
isComponentEPMOverridden	Is Component EPM overridden in problem	Boolean	N
resolvedBy	The person who resolved the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Object or null	N
createdAt	The date and time that the problem was created.	ISO 8601 date-time string	Y
lastModifiedAt	The date and time that the problem was last modified.	ISO 8601 date-time string	Y
assigneeLastModifiedAt	The date and time that the problem was modified by the assignee. For a newly assigned problem, this is set to the time of assignment.	ISO 8601 date-time string	Y
ccList	An array of CC objects for the problem, which describes the list of people who are CCed on the problem. See <a href="#">4.9.1 Get CC List</a> .	Array	N
isReadByAssignee	Indicates whether the problem has been read by the assignee.	Boolean	Y



Key	Description	Data Type	Default
isReadByProxy	Indicates whether the problem has been read by the proxy.	Boolean	N
state	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String	Y
substate	If state is not “Analyze”, this returns null. If state is “Analyze”, an enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String or null	Y
resolution	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String	Y
duplicateOfProblemID	If the problem has a resolution of “Duplicate”, this is the ID of the problem that it was marked as a duplicate of.	Integer or null	Y
classification	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String	Y
reproducible	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String	Y
milestone	The problem milestone. (See <a href="#">6.3.3 Component Milestones Included in Other Objects</a> )	Object or null	Y
priority	The priority value	Integer	Y
priorityText	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String	N
fixOrder	The fix order value	Integer	Y
taskOrder	The task order value. Float value can have max of 3 decimal values and 8 digits, but total length should not exceeds 11 characters including radix point.	Float or null	N
attachments	An array of attachment metadata objects. (See <a href="#">4.3.1 Get Problem Enclosures List</a> )	Array	N



Key	Description	Data Type	Default
pictures	An array of picture metadata objects. (See <a href="#">4.3.1 Get Problem Enclosures List</a> )	Array	N
history	The problem's history log (See <a href="#">4.11.1 Get Problem History</a> )	Array	N
diagnosis	The problem diagnosis, as an array of objects. (See <a href="#">4.2.1 Get Problem Diagnosis</a> ). Only user entered diagnosis text will be fetched.	Array	N
diagnosis.user	The user entered diagnosis text, as an array of objects	Array	N
diagnosis.history	The diagnosis history text, as an array of objects	Array	N
diagnosis.all	Both user entered and diagnosis history text as an array of objects.	Array	N
description	The problem description, as an array of objects. (See <a href="#">4.1.1 Get Problem Description</a> )	Array	N
relatedProblems	A list of related problem objects. (See <a href="#">4.4.1 Get Related Problems List</a> )	Array	N
relatedProblems.related	problems which are related-to another problem	Array of Object	N
relatedProblems.blocked-by	problems which are blocked-by other problem	Array of Object	N
relatedProblems.blocking	problems which are blocking other problem	Array of Object	N
relatedProblems.parent-of	problems which are parent-of other problem	Array of Object	N
relatedProblems.subtask-of	problems which are parent-of other problem	Array of Object	N
relatedProblems.original-of	problems which are original-of other problem	Array of Object	N
relatedProblems.duplicate-of	problems which are duplicate-of other problem	Array of Object	N
relatedProblems.clone-of	problems which are clone-of other problem	Array of Object	N



Key	Description	Data Type	Default
relatedProblems.cloned-to	problems which are clone-to other problem	Array of Object	N
targetMilestones	An array of target milestones for the problem. (See <a href="#">4.10.1 Get Target Milestones List</a> )	Array	N
targetMilestonesMasterID	The ID of the problem that is returned in Find Problem queries for the Target Milestone set. Will be null if targetMilestonesCount is zero.	Integer or null	N
otherRelatedItems	The list of “Other Related Items” for this problem. (See <a href="#">4.6.1 Get Other Related Items List</a> )	Array	N
configuration	The full configuration information. Maximum size of 1000000 characters.	String or null	N
configurationSummary	The one-line configuration summary. Maximum size of 240 characters.	String or null	Y
hasWorkaround	Indicates whether any workaround information has been filled in.	Boolean	Y
workaround	A description of the workaround. Maximum size of 1000000 characters.	String or null	N
hasSourceChanges	Indicates whether any source changes have been filled in.	Boolean	Y
sourceChanges	Source changes. Maximum size of 1000000 characters.	String or null	N
counts.thirdPartyProducts	The number of third party products related to this problem.	Integer	Y
isThirdPartyAppRelated	Indications that this problem is related to a third party app.	Boolean	N
thirdPartyProducts	The list of third party products. (See <a href="#">4.7.1 Get Third Party Product List</a> )	Array	N
keywords	An array of keyword objects. (See <a href="#">4.5.1 Get Problem Keywords List</a> )	Array	N
keywordNames	An array containing all the keyword names without its metadata.	Array of String	N



Key	Description	Data Type	Default
hasReleaseNotes	Indicates whether any release notes have been filled in.	Boolean	Y
releaseNotes	The release notes. Maximum size of 1000000 characters.	String or null	N
securityList	The list of groups or people who have access to this problem. This list will NOT include access permissions from the component. (See <a href="#">4.8.1 Get Security List</a> )	Array	N
isSecurityDefault	Returns a string describing whether individuals or groups without access to the Problem's component have been granted access to open and modify this problem (with Assignable privileges).	Enumeration	N
isExternallyViewable	Is the problem externally viewable?	Boolean	Y
adcNotes	The ADC notes for the problem. ADC notes will be returned in response only if the person is having sufficient privileges otherwise exception will be thrown.	String or null	N
failedModule	The name of the module that failed. Maximum size of 60 characters.	String or null	N
failureDetail	A one-line summary of the failure. Maximum size of 60 characters.	String or null	N
succinctSummaryRootCause	A detailed summary of the failure. Maximum size of 2000 characters.	String or null	N
actionTaken	A description of the corrective action taken. Maximum size of 2000 characters.	String or null	N
schedule	This attribute is a convenience for fetching all of the schedule-related attributes grouped together as a single object. This includes: dateNeededCurrent, dateNeededOriginal, targetCompletionCurrent, targetCompletionOriginal, targetStartDate, isApproved, isUmbrella, and isAutoCalculated.	Object	N
dateNeededCurrent	The current value for date needed. (startactualdate).	ISO 8601 date string or null	N



Key	Description	Data Type	Default
dateNeededOriginal	The original, planned value for date needed. (startplanneddate)	ISO 8601 date string or null	N
targetCompletionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date string or null	N
targetCompletionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date string or null	N
targetStartDate	The targeted start date. (duedate)	ISO 8601 date string or null	N
isApproved	Has this work been approved? (featureapproved)	Boolean	N
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean	N
isAutoCalculated	Whether to auto-calculate dates and effort from subtasks.	Boolean	N
impact	This attribute is a convenience for fetching all of the impact-related attributes grouped together as a single object. This includes: hasNewAPIImpact, hasNewSPIImpact, hasHumanInterfaceImpact, hasThirdPartyImpact, hasImportExportImpact, hasLocalizationImpact, hasPatentReviewImpact, hasConfidentialContentImpact, and hasOpenSourceImpact.	Object	N
hasNewAPIImpact	Does this problem involve new API? (featurenewapi)	Boolean	N
hasNewSPIImpact	Feature has new SPI (featurenewspi)	Boolean	N
hasHumanInterfaceImpact	Does the problem have HI impact? (featurehi)	Boolean	N
hasThirdPartyImpact	Feature has third party content (feature3rdparty)	Boolean	N
hasImportExportImpact	Does this problem require import / export review? (featureimpexp)	Boolean	N
hasLocalizationImpact	Does the problem have localization impact? (featureloc)	Boolean	N



Key	Description	Data Type	Default
hasPatentReviewImpact	Feature needs patent review (featureneedspatent)	Boolean	N
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean	N
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean	N
effort	This attribute is a convenience for fetching all of the effort-related attributes grouped together as a single object. This includes: effortCurrentTotalEstimate, effortOriginalTotalEstimate, effortPercentComplete, effortRemaining, and effortExpended.	Object	N
effortCurrentTotalEstimate	Effort, current total estimate, in days. (effortcurtotal)	Float or null	N
effortOriginalTotalEstimate	Effort, original total estimate, in days. (effortinittotal)	Float or null	N
effortPercentComplete	Effort, percent complete, from 0 to 100. (effortpercentcomplete)	Integer or null	N
effortRemaining	Effort, remaining, in days. (effortremain)	Float or null	N
effortExpended	Effort, expended, in days. (effortexpended)	Float or null	N
testCase	Feature test case (featuretextcaseid). Maximum size of 768 characters.	String or null	N
foundInBuild	The name of the component build that this problem was found in. Maximum size of 25 characters.	String or null	N
fixedInBuild	The name of the component build that this problem was fixed in. Maximum size of 25 characters.	String or null	N
verifiedInBuild	The name of the component build that this problem was verified in. Maximum size of 25 characters.	String or null	N
mustBeFixedInBuild	The name of the component build that this problem must be fixed in. Maximum size of 25 characters.	String or null	N





Key	Description	Data Type	Default
isVerifiedByTester	Was this problem verified by a tester?	Boolean	N
isRegressionRequired	Must this problem be regressed?	Boolean	N
resolvedAt	The date and time that the issue was resolved	ISO 8601 date-time string or null	N
closedAt	The date and time that the issue was closed	ISO 8601 date-time string or null	N
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Maximum size of 2048 characters.	String or null	N
componentID	ID of component to which problem belongs	Integer	N
stateCode	State code for problem	Integer	N
subStateCode	SubState Code for problem	Integer	N
resolutionCode	Resolution Code for problem	Integer	N
event	The problem event name.	Event object or null	N
productSecurity	Product Security Object	ProductSecurity Object	N
label	Label details of the problem. See <a href="#">Response Attributes</a> table in the section 11.1 Get Labels, for description.	Label Object or null	N
recentAssignees	Array of recent assignees object mentioned below	Array of recent assignee object	N
relatedTests	A list of related tests objects. Refer 4.19.1 Get Related Tests Object attributes	Array of related Test Object	N
counts	An object containing counts of problem properties. Counts object is defined in table 1.4.	Object	N

## Recent Assignee Object Attribute

Key	Description	Data Type
person	Person object contain previous assignee details	Person Object



Key	Description	Data Type
type	A string that represents the type of assignee	Enumerated String

### 1.1 ProductSecurity Object

Key	Description	Data Type
type	SecurityType code of the product security. ENUMERATED VALUE	String
isPrivacyIssue	Does this product contain a privacy issue.	Boolean or null
cwe	CWE (Not defined). CWE object is defined in below table 1.2	CWE Object
color	Color code of the problem. ENUNMERATED VALUE	String
isColorOverridden	Is the color overridden by user	Boolean
vulnerabilityClassification	An object containing vulnerability information. Defined in table 1.3	Vulnerability Object
visibility	Visibility of the issue. ENUMERATED VALUE	String
securityDRI	Person object of the Security DRI	Object or null
securityVerifier	Person object of the Security Verifier	Object or null
securityMasterID	Security Master ID for the product security	Integer
foundAt	Date found of the oldest bug in Security umbrella	ISO 8601 date string without null
disclosedAt	Disclosure date of the Product Security	ISO 8601 date string without null
targets	Array of targets object. See <a href="#">Response Attributes</a> table in the section 4.13.1 Get Targets for problem for detail.	Array of Target Object
reporters	Array of reporters object. See <a href="#">Response Attributes</a> table in the section 4.14.1 Get Reporters for problem for detail.	Array of Reporter object
externals	Array of externals object. See <a href="#">Response Attributes</a> table in the section 4.15.1 Get Externals for problem for detail.	Array of External objects

### 1.2 CWE Object



Key	Description	Data Type
primary	List Provided by Product Security Team	String or null
secondary	List Provided by Product Security Team	String or null

### 1.3 Vulnerability Object

Key	Description	Data Type
attackVector	The attack vector. ENUMERATED VALUE	String
authentication	Authentication for the Product Security object. ENUMERATED VALUE	String
assets	Assets code of the Product Security. ENUMERATED VALUE	String
complexity	Complexity of the problem. ENUMERATED VALUE	String
exploitability	Exploitability of the problem. ENUMERATED VALUE	String
impact	Impact of the problem. ENUMERATED VALUE	String
userBase	User base of the problem. ENUMERATED VALUE	String

### 1.4 Counts Object

Key	Description	Data Type
attachments	The number of attachments.	Integer
pictures	The number of pictures.	Integer
cc	The number of people who are CCed on the problem.	Integer
relatedProblems	The number of related problems	Integer
targetMilestones	The number of target milestones for a problem	Integer
otherRelatedItems	The number of "Other Related Items"	Integer



Key	Description	Data Type
thirdPartyProducts	The number of third party products related to this problem.	Integer
keywords	The number of keywords associated with the problem.	Integer
securityList	The number of items on the securityList. Zero entries corresponds to "default" security.	Integer
duplicates	Number of duplicate related problems	Integer
blocking	Number of Blocking related problems	Integer
blockedBy	Number of Blocked-by related problems	Integer
crashes	The problem crash count	Integer
relatedTests	The number of related Tests	Integer
seedUsers	The number of Seed Users	Integer

### [E] Examples

#### Get a Single Problem:

Client request:

```
GET /problems/9000000
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.1
Status: 200
Content-Type: application/json; charset=utf-8
{
  "hasSourceChanges": false,
  "reproducible": "I Didn't Try",
```



```
"state": "Analyze",
"resolution": "Unresolved",
"id": 9000000,
"milestone": {
  "component": {
    "name": "Mac OS",
    "version": "X"
  },
  "name": "Later"
},
"fingerprint": "5f38548e",
"title": "Foldable Mac Book Air",
"isExternallyViewable": false,
"component": {
  "name": "Mac OS",
  "version": "X"
},
"createdAt": "2011-02-14T21:50:17+0000",
"priority": 2,
"lastModifiedAt": "2014-01-17T07:26:19+0000",
"hasReleaseNotes": false,
"isReadByAssignee": false,
"fixOrder": 6,
"assignee": {
  "lastName": "Uchida",
  "email": "uchida@apple.com",
  "dsid": 1433115649,
  "firstName": "Nicholas",
  "type": "Employee"
},
"assigneeLastModifiedAt": "2014-01-17T07:26:19+0000",
"substate": "Screen",
"duplicateOfProblemID": null,
"configurationSummary": null,
"classification": "Task",
"hasWorkaround": false,
"counts": {
  "securityList": 1,
  "duplicates": 0,
  "keywords": 5,
  "pictures": 3,
  "seedUsers": 0,
  "targetMilestones": 0,
  "relatedTests": 0,
  "crashes": 0,
  "blockedBy": 0,
  "blocking": 0,
  "thirdPartyProducts": 0,
  "relatedProblems": 18,
  "otherRelatedItems": 0,
  "attachments": 0,
  "cc": 92
},
"originator": {
  "lastName": "Vaithyanathan",
```



```

        "email": "kartik@apple.com",
        "dsid": 187179976,
        "firstName": "Kartik",
        "type": "Employee"
    }
}

```

### Specifying a list of fields

The fields requested in X-Fields-Requested should override the default set.

Client request:

```

GET /problems/9000000
X-API-Version: 1.0
X-Fields-Requested: title,component,state,substate,resolution,
priority,relatedProblems

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "component": {
    "name": "Mac OS",
    "version": "X"
  },
  "priority": 2,
  "relatedProblems": [
    {
      "problem": {
        "assignee": {
          "dsid": 3795,
          "email": null,
          "firstName": "Brian",
          "lastName": "Lewis",
          "type": "Employee"
        },
        "component": {
          "name": "Alaska 98000",
          "version": "3.0"
        },
        "id": 2000000,
        "state": "Closed",
        "title": "Assigning the first bug!!!"
      },
      "relationType": "related to"
    },
    ...
  ],
  "resolution": "Unresolved",
  "state": "Analyze",
  "substate": "Screen",

```



```

    "title": "Foldable Mac Book Air"
  }

```

### Example for Product Security Object

Client request:

```

GET /problems/10052528
X-Fields-Requested: id,productSecurity,epm,isComponentEPMOVERRIDDEN

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
{
  "epm": {
    "dsid": 9061211,
    "email": "mlock@apple.com",
    "firstName": "Michael G. L.",
    "lastName": "Lock",
    "type": "Employee"
  },
  "id": 9000000,
  "isComponentEPMOVERRIDDEN": true,
  "productSecurity": {
    "color": "Blue",
    "cwe": {
      "primary": "Race Condition",
      "secondary": "Race Condition"
    },
    "disclosedAt": "2011-12-21",
    "externals": [],
    "foundAt": "2011-12-21",
    "isColorOverridden": false,
    "isPrivacyIssue": null,
    "reporters": [
      {
        "isCoordinate": true,
        "isCredit": false,
        "email": "john@smith.com",
        "id": 123,
        "name": "John Smith",
        "isSeed": false,
        "sonar": "1234"
      },
      {
        "isCoordinate": null,
        "isCredit": false,
        "email": "newt@smith.com",
        "id": 124,
        "name": "Newt Smith",
        "isSeed": true,
        "sonar": "1234"
      }
    ]
  }
}

```



```
"securityDRI": {
  "dsid": 9061211,
  "email": "mlock@apple.com",
  "firstName": "Michael G. L.",
  "lastName": "Lock",
  "type": "Employee"
},
"securityMasterID": 9000000,
"securityVerifier": {
  "dsid": 9061211,
  "email": "mlock@apple.com",
  "firstName": "Michael G. L.",
  "lastName": "Lock",
  "type": "Employee"
},
"targets": [
  [
    {
      "component": {
        "name": "Radar",
        "version": "7.0"
      },
      "id": 123,
      "isCriticalToFix": false,
      "plannedReleaseVehicle": {
        "name": "Radar-7.0.2",
        "id": 421
      },
      "affectedProduct": {
        "name": "Radar-7.0",
        "id": 410
      },
      "recommendedReleaseVehicle": {
        "name": "Radar-7.1",
        "id": 434
      },
      "trackingProblemID": null,
      "isVerified": null
    },
    {
      "component": {
        "name": "Radar",
        "version": "7.1"
      },
      "id": 124,
      "isCriticalToFix": true,
      "plannedReleaseVehicle": {
        "name": "Radar-7.1.1",
        "id": 435
      },
      "affectedProduct": {
        "name": "Radar-7.1",
        "id": 425
      },
      "recommendedReleaseVehicle": {
```





```

        "name": "Radar-7.1",
        "id": 434
        "trackingProblemID": null,
        "isVerified": false
    }
    ]
    ],
    "type": "Mitigation",
    "visibility": "Not Set",
    "vulnerabilityClassification": {
        "assets": "General",
        "attackVector": "Local system",
        "authentication": "Anonymous",
        "complexity": "Straightforward",
        "exploitability": "Theoretical",
        "impact": "Total compromise",
        "userBase": "Most systems"
    }
}
}
}

```

#### Example of related-problems List in X-Fields-Requested

Client request:

```

GET /problems/12321710
"X-Fields-Requested: id,relatedProblems.related-to,relatedProblems.blocked-
by,relatedProblems.blocking,relatedProblems.parent-
of,relatedProblems.subtask-of,relatedProblems.original-
of,relatedProblems.duplicate-of,relatedProblems.clone-
of,relatedProblems.cloned-to"

```

Response:

```

HTTP/1.1 200 OK
Date: Wed, 22 May 2013 06:13:29 GMT
X-API-Version: 1.2
Content-Type: application/json;charset=UTF-8

```

```

{
  "id": 12321710,
  "relatedProblems.original-of": [
    {
      "id": 12309783,
      "title": "Test the problem - b16",
      "component": {
        "name": "TestPrivi",
        "version": "7.2"
      },
      "state": "Analyze",
      "assignee": {
        "lastName": "Tester1",

```



```

        "email": "radartester01@gmail.com",
        "dsid": 1118580968,
        "type": "Contractor",
        "firstName": "radar"
    }
}
],
"relatedProblems.related-to": [
    {
        "id": 80904,
        "title": "Read causes hang",
        "component": {
            "name": "Serial Tool",
            "version": "1.0.1"
        },
        "state": "Analyze",
        "assignee": {
            "lastName": "TBH",
            "email": " ",
            "dsid": -28112,
            "type": "No Access",
            "firstName": "System Update"
        }
    },
    {
        "id": 102460,
        "title": "test?srda",
        "component": {
            "name": "A/UX",
            "version": "3.0"
        },
        "state": "Closed",
        "assignee": {
            "lastName": "Pavelko",
            "email": "pavelko@apple.com",
            "dsid": 1834,
            "type": "No Access",
            "firstName": "Paul"
        }
    }
],
"relatedProblems.duplicate-of": [
    {
        "id": 12311819,
        "title": "CrashTracer: [USER] Calendar at com.apple.AppKit: -
[NSApplication _crashOnException:] + 106 :: Collection <__NSArrayM: ADDRESS>

```



```
"component": {  
    "name": "Calendar (New Bugs)",  
    "version": "X"  
},  
"state": "Analyze",  
"assignee": {  
    "lastName": "Tester1",  
    "email": "radartester01@gmail.com",  
    "dsid": 1118580968,  
    "type": "Contractor",  
    "firstName": "radar"  
}  
  
},  
],  
"relatedProblems.cloned-to": [  
    {  
        "id": 12321733,  
        "title":  
"1232171512321715123217151232171512321715123217151232171512321715123  
2171512321715123217151232171512321715123217151232171512321715123217151232171  
5123217151232171512321715123217151232171512321715123217151232171512321715123  
2171512321715",  
        "component": {  
            "name": "Accessory Interface Specs",  
            "version": "Dropbox"  
        },  
        "state": "Analyze",  
        "assignee": {  
            "lastName": "Foo",  
            "email": "efoo@apple.com",  
            "dsid": 1068920402,  
            "type": "Employee",  
            "firstName": "Edwin"  
        }  
    }  
],  
"relatedProblems.blocked-by": [  
    {  
        "id": 11240611,  
        "title": "Testing Mac pouncer usage in production",  
        "component": {  
            "name": "Calendar (New Bugs)",  
            "version": "X"  
        },  
        "state": "Verify",
```



```

    "assignee": {
      "lastName": "ADCBugs",
      "email": null,
      "dsid": 17856935,
      "type": "Contractor",
      "firstName": "Developer"
    }
  },
  ],
  "relatedProblems.parent-of": [
    {
      "id": 1231787,
      "title": "\"Keys out of order\" problem found - DFA 7.2 based
code doesn't find problem",
      "component": {
        "name": "Disk First Aid",
        "version": "7.2.1"
      },
      "state": "Closed",
      "assignee": {
        "lastName": "Brady",
        "email": "brady@apple.com",
        "dsid": 4725,
        "type": "No Access",
        "firstName": "Don"
      }
    }
  ],
  "relatedProblems.blocking": [
    {
      "id": 12311615,
      "title": "wqwSELECT COUNT(*) FROM tablenameSELECT COUNT(*) FROM
tablenameSELECT COUNT(*) FROM tablenameSELECT COUNT(*) FROM tablenameSELECT
COUNT(*) FROM tablenameSELECT COUNT(*) FROM tablenameSELECT COUNT(*) FROM
tablenameSELECT COUNT(*) FROM tablenameSELECT COUNT(",
      "component": {
        "name": "Radar",
        "version": "7.2"
      },
      "state": "Verify",
      "assignee": {
        "lastName": "Tester1",
        "email": "radartester01@gmail.com",
        "dsid": 1118580968,
        "type": "Contractor",
        "firstName": "radar"
      }
    }
  ]

```



```

    }
  ],
  "relatedProblems.subtask-of": [
    {
      "id": 102451,
      "title": "auto 11593.698: Color box appears with grid or stripe
in highlight popup\n(Color",
      "component": {
        "name": "Color Picker Package",
        "version": "Psycho Fmr"
      },
      "state": "Closed",
      "assignee": {
        "lastName": "Reak",
        "email": "reak.c@apple.com",
        "dsid": 10616,
        "type": "No Access",
        "firstName": "Casey"
      }
    }
  ],
  "relatedProblems.clone-of": [
    {
      "id": 12309783,
      "title": "Test the problem - b16",
      "component": {
        "name": "TestPrivi",
        "version": "7.2"
      },
      "state": "Analyze",
      "assignee": {
        "lastName": "Tester1",
        "email": "radartester01@gmail.com",
        "dsid": 1118580968,
        "type": "Contractor",
        "firstName": "radar"
      }
    }
  ]
}

```

**Example of date format shown for date only fields and date with time and timeZone fields**

Client request:

GET /problems/12321710



"X-Fields-Requested: id, dateNeededCurrent, dateNeededOriginal, createdAt, lastModifiedAt"

**Response:**

HTTP/1.1 200 OK

Date: Wed, 22 May 2013 06:13:29 GMT

X-API-Version: 1.2

Content-Type: application/json; charset=UTF-8

```
{
  "id": 123,
  "dateNeededCurrent": "2013-05-08",
  "dateNeededOriginal": null,
  "createdAt": "2013-02-08T06:40:55+0000",
  "lastModifiedAt": "2013-06-06T08:51:01+0000",
}
```

**Example of counts fields with cc**

Client request:

GET /problems/12321710

"X-Fields-Requested: id, counts.cc"

**Response:**

HTTP/1.1 200 OK

Date: Wed, 22 May 2013 06:13:29 GMT

X-API-Version: 1.6

Content-Type: application/json; charset=UTF-8

```
{
  "counts": {
    "cc": 0,
  },
  "id": 16156099
}
```

**Example of counts fields**

Client request:

GET /problems/12321710

"X-Fields-Requested: id, counts"

**Response:**

HTTP/1.1 200 OK

Date: Wed, 22 May 2013 06:13:29 GMT

X-API-Version: 1.6



```
Content-Type: application/json;charset=UTF-8
{
  "id": 16156099,
  "counts": {
    "securityList": 2,
    "duplicates": 0,
    "keywords": 0,
    "pictures": 6,
    "targetMilestones": 1,
    "seedUsers": 0,
    "crashes": 0,
    "relatedTests": 0,
    "blockedBy": 0,
    "blocking": 0,
    "thirdPartyProducts": 0,
    "relatedProblems": 3,
    "otherRelatedItems": 0,
    "attachments": 21,
    "cc": 0
  }
}
```

## 3.2 Find Problems

### [A] Description

This API is used to find problems via a JSON POST interface. The client can specify a JSON hash that contains the fields against which to perform a search, including what kind of comparison operator to use per field.

Custom fields can also be requested using the `X-Fields-Requested` header, or supplying a `"fieldsRequested"` field in the request.

If the `"idsOnly"` or `"countsOnly"` fields are supplied and set to boolean `"true"`, the result set will contain the list of problem IDs or count of the problem IDs, respectively.

The response of `"idsOnly"` field will contain array of ids.

The default number of rows returned is 2000. You may increase the number of rows returned by using the `"X-rowlimit"` header. Note that a large rowlimit value with too broadly defined query criteria can lead to a connection timeout. If this happens, focus your criteria more narrowly.

The ability to paginate the search results is not supported.



## [B] Schedule

For version 1.0, the list of searchable fields should include everything supported by the present day RadarCLI GetProblemData. It is not required for version 1.0 to supported deep searches into diagnosis, description, or other fields presently returned by GetProblemText.

## [C] URL Scheme

POST /problems/find

## [D] Request Attributes

The following problem attributes can be searched:

Key	Data Type
id	Integer
includeInactive	Boolean
idsOnly	Boolean
countsOnly	Boolean
createdAt	Date/time
lastModifiedAt	Date/time
assigneeLastModifiedAt	Date/time
title	String
isReadByAssignee	Boolean
isInMyCCList	Boolean
component	Component
componentBundle	Component Bundle
state	Enumerated String
substate	Enumerated String
resolution	Enumerated String
milestone	String or null
fixedInBuild	String or null
foundInBuild	String or null





Key	Data Type
mustBeFixedInBuild	String or null
verifiedInBuild	String or null
priority	Integer
taskOrder	Float
fixOrder	Integer
keyword	Keyword
assignee	Person
originator	Person
counts	Counts Object
event	String or null
productSecurity	Product SecurityObject
resolvedAt	Date/ time
closedAt	Date/ time
resolvedBy	Person
lastModifiedBy	LastModifiedBy Object
label	Label Object
hasDefaultSecurity	Boolean
isVerifiedByTester	Boolean
otherRelatedItem	OtherRelatedItem Object
classification	Enumerated String
reproducible	Enumerated String
dri	Person
epm	Person
componentHistory	Component
buildInfo	String

#### OtherRelatedItem Object



Key	Data Type
system	String
url	String
title	String

### ProductSecurity Object

Key	Data Type
type	Integer
isPrivacyIssue	Boolean or null
color	Enumerated String
visibility	Enumerated String
securityDRI	Person
securityVerifier	Person
securityMasterID	Integer
foundAt	Date/Time
disclosedAt	Date/Time
targetsCount	Integer
reportersCount	Integer
externalsCount	Integer
isProblemSecurityMaster	Boolean
cwe	CWE Object
vulnerabilityClassification	VulnerabilityClassification Object
targets	Target Object
reporters	Reporter Object
externals	External Object

### CWE Object



Key	Data Type
primary	String
secondary	String

**VulnerabilityClassification Object**

Key	Data Type
attackVector	Enumerated String
authentication	Enumerated String
assets	Enumerated String
complexity	Enumerated String
exploitability	Enumerated String
impact	Enumerated String
userBase	Enumerated String

**Target Object**

Key	Data Type
affectedProduct	String
recommendedReleaseVehicle	String
plannedReleaseVehicle	String
trackingProblem	Integer
isCriticalToFix	Boolean
isVerified	Boolean or null

**Reporter Object**

Key	Data Type
name	String
email	String



Key	Data Type
sonar	String
isSeed	Boolean or null
isCredit	Boolean or null
isCoordinate	Boolean or null

**External Object**

Key	Data Type
external	String

**Label Object**

Key	Data Type
name	String
setName	String

**LastModifiedBy Object**

Key	Data Type
dsid	Integer
datesBetween	Date

**Counts Object**

Key	Data Type
crash	Integer

The domain of search values that can be specified for an attribute is dependent on the attribute's data type. The following table describes the list of values that can be specified without the use of operators.



Data Type	Possible Values
Integer	An integer or list of integers. With a list, the result will be problems that match ANY of the supplied values.
Float	An integer or list of integers. With a list, the result will be problems that match ANY of the supplied values.
Boolean	true or false
Date/time	An ISO 8601 date-time string or date string, or a list of date strings. When a date string is supplied, the search ranges from 00:00 to 24:00 GMT hours on the given date. With a list, the result will be problems that match ANY of the supplied dates.
String	An exact string or list of exact strings to match. With a list, the result will be problems that match ANY of the supplied values.
Enumerated String	One of the enumerated string values, or a list of such values. With a list, the result will be problems that match ANY of the supplied values.
Component	An component specified as an object with "name", "version" and optional "includeSubcomponents" Key.
Component Bundle	An component bundle specified as a string name. If a private and global component bundle both have the same name, the private bundle will be used.
Keyword	A keyword specified as a string name or an integer ID, or a list of such values. Since keywords are a collection attribute, with a list, the result will be problems that match ALL of the supplied values.
Person	A person specified as an integer DSID, or a list of DSIDs. With a list, the result will be problems that match ANY of the supplied values.
OtherRelatedItem	A other related item based problem search. With a list, the result will be problems that match ANY of the supplied values.

For example, this search will find all problems in Analyze:

```
POST /problems/find
{
  "state": "Analyze"
}
```

And this search will find all problems that are either in Analyze or in Verify:

```
POST /problems/find
{
  "state": [
    "Analyze",
    "Verify"
  ]
}
```



```
    ]
  }
```

The following search will find all problems that have been tagged with every keyword in the list:

```
POST /problems/find
{
  "keyword": [
    "Snakes on a Radar",
    "OSX-Perf Reviewed"
  ]
}
```

### Option Attributes

Some request attributes are not fields upon which to search, but are treated as options that modify the search results.

- "idsOnly" causes the request to return an array of problem IDs. It is mutually exclusive with "countsOnly" and "fieldsRequested".
- "countsOnly" causes the request to return a simple object whose sole key "count" has a integer value representing the number of matching problem IDs. It is mutually exclusive with "idsOnly" and "fieldsRequested".
- "includeInactive" extends the search to inactive radars.
- "fieldsRequested" specifies the set of fields to return for each problem. It is mutually exclusive with "idsOnly" and "countsOnly".
- A custom "additionalWhereClause" may also be specified (although we should go to some pains to downplay this, or perhaps not actually document it).

### Attribute Operators

Instead of a field value or list of field values, a search can be specified as an object whose Key are arguments. The list of possible arguments are:

Operators	Description
eq	Equality: this is the same as not using an operator.
neq	Not equal
gt	Greater than
gte	Greater than or equal
lt	Less than
lte	Less than or equal
any	The attribute being searched must contain ANY of the values in the list supplied.



Operators	Description
all	The attribute being searched must contain ALL of the values in the list supplied.
none	The attribute being searched must contain NONE of the values in the list supplied.
like	The attribute being searched must match a wildcard search, with a leading or trailing "%" as the operator.

This table lists the operators that apply to each data type:

Data Type	Supported Operators
Integer	eq, neq, gt, gte, lt, lte, any, none
Float	eq, neq, gt, gte, lt, lte, any, none
Date/time	eq, neq, gt, gte, lt, lte, any, none
String	eq, neq any, none, like
Enumerated String	eq, neq, gt, gte, lt, lte, any, none (Ordering is specified by the list of enumerated values, not alphabetically. <a href="#">See 10.1 Get Field Enumeration</a> )
Component	eq, neq, any, none
ComponentBundle	eq, neq, any, none
Keyword	eq, neq, any, all, none
Person	eq, neq, any, none
OtherRelatedItem	eq, any

Problems whose priority is in the set P1, P2, P3.

```
"priority": {
  "any": [
    1,
    2,
    3
  ]
}
```

Problems whose keywords have **any members** of this set.

```
"keyword": {
  "any": [
    "Angry Croatian Hitlist",
    "Snakes on a Radar"
  ]
}
```



```
}
```

Problems whose keywords have **all members** of this set.

```
{
  "keyword": {
    "all": [
      "Angry Croatian Hitlist",
      "Snakes on a Radar"
    ]
  }
}
```

Problems whose priority **is not** in the set P1, P2, P3.

```
"priority": {
  "none": [
    1,
    2,
    3
  ]
}
```

Problems whose keywords contain **none of the members** of this set.

```
"keyword": {
  "none": [
    "Angry Croatian Hitlist",
    "Snakes on a Radar"
  ]
}
```

Some of the operators can be used in combinations to effect range or set searches:

Problems created after 2012-01-01 23:59:59 UTC, but before or equal to 2012-03-01 00:00:00 UTC.

```
"createdAt": {
  "gt": "2012-01-01T23:59:59+0000",
  "lte": "2012-03-01T00:00:00+0000"
}
```

Problems that have “Angry Croatian Hitlist” or “Snakes on a Radar” but not “Program Hitlist”.

```
"keyword": {
  "any": [
    "Angry Croatian Hitlist",
    "Snakes on a Radar"
  ],
  "none": [
    "Program Hitlist"
  ]
}
```

Problems with the string “crumpets” in the title (always ignoring case sensitivity) and who are also tagged with “Angry Croatian Hitlist” or “Snakes on a Radar” keywords.

```
POST /problems/find
{
  "keyword": {
```





```

    "any": [
        "Angry Croatian Hitlist",
        "Snakes on a Radar"
    ],
    "title": {
        "like": "%crumpets%"
    }
}

```

Search for problems created between the beginning of January 1, PST, and the end of January 2, PST:

```

{
    "createdAt": {
        "gte": "2012-01-01T00:00:00-0800",
        "lte": "2012-01-02T00:00:00-0800"
    }
}

```

### Component Search

‘Find Problem’ request for component Object will now by default will support for ‘equal’ operator instead of ‘like’ operator. (as per radar <[rdar://problem/13024778](http://rdar://problem/13024778)>)

For Example -

```

{
    "component": {
        "name": "Pep",
        "version": "%"
    }
}

```

The above request will search for component with exact match of ‘%’ instead of using it as wild card. If user wants to search as a wild card, then request should be passed as below-

```

{
    "component": {
        "name": "Pep",
        "version": {
            "like": "%"
        }
    }
}

```

### Problem History Search

The “everAssignedTo” key can be used to search for problems that have ever been assigned to a component or a person (assignee)



For example, this search will return all problems that were ever assigned to the “Alaska”/“Bloom Cnty” component:

```
POST /problems/find
{
  "everAssignedTo": {
    "component": {
      "name": "Alaska",
      "version": "Bloom Cnty"
    }
  }
}
```

This search will return all problems that were ever assigned to a particular user:

```
POST /problems/find
{
  "everAssignedTo": {
    "assignee": 107254372
  }
}
```

All of the same rules apply to history attribute searches as to regular searches, including lists, operators, etc.

#### **‘organizationOf’ and ‘directReportOf’ Search**

The ‘organizationOf’ and ‘directReportOf’ attribute search is supported for assignee, originator, dri, securityVerifier, SecurityDRI, resolvedBy person based problem search.

Problems whose assignee is in organizationOf can be search with below request attribute. If ‘organizationOf’ key contains DSID as a value then it will be equal search.

```
"assignee":
{
  "organizationOf": 1118580968
}
```

```
"assignee":
{
  "directReportOf": 1118580968
}
```

The supported operators for this search is ‘eq’, ‘neq’, ‘any’ and ‘none’.

```
"assignee":
{
  "organizationOf": {"eq": 1118580968}
}
```

Problems whose assignee is not in organizationOf can be search with below request attribute.

```
"assignee":
```



```
{
  "organizationOf": {"neq": 1118580968}
}
```

Problems whose assignee is in any of organizationOf can be search with below request attribute.

```
{
  "organizationOf": {"any": [1118580968, 8867, 154647386]}
}
```

### **‘radarWebProgram’ in originator based problem search**

A new attribute ‘radarWebProgram’ is supported for originator in Find problem API. If ‘radarWebProgram’ with some program name used then problems whose originator is under given Originator Program will be returned as response. RadarWeb program passed in request will be validated and if unique then only it will be used in search criteria of find problem API. If RadarWeb program not found in DB then appropriate error message will be shown.

```
"originator":
{
  "radarWebProgram": "Radar Training"
}
```

The supported operators for this search is ‘eq’, ‘neq’, ‘any’ and ‘none’. Default operator is ‘eq’.

```
"originator":
{
  "radarWebProgram": {"eq": "Radar Training"}
}
```

Problems whose originator is not in radarWebProgram can be search with below request attribute.

```
"originator":
{
  "radarWebProgram": {"neq": "Radar Training"}
}
```

Problems whose originator is in any of radarWebProgram can be search with below request attribute.

```
"originator":
{
  "radarWebProgram": {"any": ["Radar Training", "Radarweb Test
Program"]}
```

### **‘radarGroup’ and ‘dsGroup’ Search**

The ‘radarGroup’ and ‘dsGroup’ attribute search is supported for assignee, originator, dri, securityVerifier, SecurityDRI, resolvedBy person based problem search.

Problems whose assignee is in radarGroup ‘Developers’ can be search with below request attribute. If ‘radarGroup’ key contains groupName as a value then it will be equal search.

```
"assignee":
```



```
{
  "radarGroup": "Developers"
}
```

The supported operators for this search is 'eq', 'neq', 'any' and 'none'.

```
"assignee":
{
  "dsGroup": {"eq": "Developers"}
}
```

Problems whose assignee is not in 'Developers' can be search with below request attribute.

```
"assignee":
{
  "radarGroup": {"neq": "Developers"}
}
```

Problems whose assignee is in any of 'Developers' can be search with below request attribute.

```
"assignee":
{
  "radarGroup": {"any": ["Developers", "Vendors"]}
}
```

### **LastModifiedBy Search**

Example for lastModifiedBy object -

```
{
  "lastModifiedBy": {
    "dsid": 1118580968,
    "datesBetween": {
      "from": "2013-05-20",
      "to": "2013-05-24"
    }
  }
}
```

In the above request body, dsid is mandatory and datesBetween attribute is optional. If datesBetween attribute is mentioned then 'from' date is mandatory. Date mentioned in 'from' should be lesser than 'to' otherwise an error message will be thrown.

LastModifiedBy attribute supported operators are 'eq', 'neq', 'any' and 'none'. It also support 'organizationOf' and 'directReportOf' based search.

```
{
  "lastModifiedBy": {
    "eq": {
      "dsid": 1118580968,
```



```

        "datesBetween": {
            "from": "2013-05-20",
            "to": "2013-05-24"
        }
    }
}

{
    "lastModifiedBy": {
        "organizationOf": {
            "dsid": 1118580968,
            "datesBetween": {
                "from": "2013-05-20",
                "to": "2013-05-24"
            }
        }
    }
}

```

#### Search problems by Label:-

Problems can be searched based on label assign to it. Below Request will give all the problems having label as 'Important' and in Label set 'RadarWS 1.3'

```

{
    "label": {
        "name": "Important",
        "setName": "RadarWS 1.3"
    }
}

```

To search for all problem having any label from a label set 'RadarWS 1.3'

```

{
    "label": {
        "setName": "RadarWS 1.3"
    }
}

```

#### Search with any and all operators to do a union and intersection operations on any attributes

To search for all the problems which is having state as 'Analyze' and belongs to either component 'Radar | Automation' or having priority as 1.

```

{
    "all": {
        "state": "Analyze",
        "any": {
            "priority": 1,
            "component": { "name": "Radar", "version": "Automation" }
        }
    }
}

```



```

    }
  }
}

```

To search problem having milestone like "01\ /31\ /14" and any of (state equal to Analyze and assignee equal to 1411775887) or (state not equal to Analyze and resolvedBy equal to 1411775887)

```

{
  "milestone": {
    "like": "01/31/14"
  },
  "any": [
    {
      "all": {
        "state": {
          "eq": "Analyze"
        },
        "assignee": {
          "eq": 1411775887
        }
      }
    },
    {
      "all": {
        "state": {
          "neq": "Analyze"
        },
        "resolvedBy": {
          "eq": 1411775887
        }
      }
    }
  ]
}

```

### Find Problem with orderBy to sort response data

To sort the find problem response based on some attributes 'orderBy' attribute can be used, which will sort the response as per field and order passed in 'orderBy' object. It will accept maximum of three order by attribute object in single request, if it exceeds defined limit then error message will be shown.

Parameter	Description	Data Type
orderBy	orderBy will be an Array of OrderBy Objects. Max of three OrderBy Object will be supported to overcome obscene amount of orderBy condition	Array of OrderByObject

### OrderBy Object



Parameter	Description	Data Type
field	The column Name to order by. Enumeration Find Problem OrderBy fields has been mentioned in enumeration section.	Enumerated values
order	The sort order. Supported values are ascending and descending	Enumerated String

In order to sort response parameter with problemID and descending, below format need to be used.

```
POST /problems/find
{
  "component": {
    "name": "SUITools",
    "version": "X"
  }
  "orderBy": [
    { "field": "id", "order": "descending" }
  ]
}
```

### Find Problem with Other Related Item Object

Other Related Item will be an object of three optional attributes 'system', 'url', 'title'. Other Related Item object should contain any one of mentioned three optional attributes otherwise an error message will be shown saying empty object.

The attribute 'system' must contain the name of valid systems which can be found using 4.6.2 Get Other Related Items Systems List API. If an invalid or non-registered system name is passed then an error message will be shown.

Supported operators for Other Related Item object are 'eq' and 'any'. Operators will not be supported inside Other Related Item object.

Sample example of Other Related Item object with supported operators are as follows

1. Other Related Item object with 'eq' operator

```
{
  "otherRelatedItem": {
    "eq": {
      "system": "Espresso",
      "url": "<exp2://Ticket/17952942>"
    }
  }
}
```

2. Direct Other Related Item object which is similar to above object with 'eq' operator

```
{
  "otherRelatedItem": {
```



```

    "system": "Espresso",
    "url": "<exp2://Ticket/17952942>"
  }
}

```

### 3. Multiple Other Related Item with 'any' operator

```

{
  "otherRelatedItem": {
    "any": [
      {
        "system": "Espresso",
        "url": "<exp2://Ticket/17952942>"
      },
      {
        "system": "Radar",
        "url": "<rdar://problem/12754889>"
      }
    ]
  }
}

```

### 4. Direct Other Related Item Array which is similar to above object with 'any' operator

```

{
  "otherRelatedItem": [
    {
      "system": "Espresso",
      "url": "<exp2://Ticket/17952942>"
    },
    {
      "system": "Radar",
      "url": "<rdar://problem/12754889>"
    }
  ]
}

```

### Find Problem with counts Object

```

{
  "counts": {
    "crash": 10
  }
}

```

### [E] Response Attributes

Below is the list of default response parameters for FindProblem Request.

Key	Description	Data Type	Default
id	Unique ID of the problem	Integer	Y





Key	Description	Data Type	Default
title	Title of the problem. Max Length is 240 Characters	String	Y
component	The component that the problem belongs to. (See <a href="#">6.5 Components Included in Other Objects</a> )	Component Object	Y
assignee	The person who is assigned to the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Person Object	Y
lastModifiedAt	The date and time that the problem was last modified.	ISO8601 Date String	Y
state	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String	Y
substate	If state is not "Analyze", this returns null. If state is "Analyze", an enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String or null	Y
classification	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String	Y
milestone	The problem milestone. (See <a href="#">6.3.3 Component Milestones Included in Other Objects</a> )	Milestone Object or null	Y
priority	The priority value	Integer	Y
fixOrder	The fix order value	Integer	Y
fingerprint	A string that the server uses to validate that the problem is the most recent version. It is a Hash Encoded value of lastModifiedAt.	String	Y
isReadByAssignee	Indicates whether the problem has been read by the assignee.	Boolean	N
isReadByProxy	Indicates whether the problem has been read by the proxy.	Boolean	N
resolution	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String	N



Key	Description	Data Type	Default
resolvedBy	The person who resolved the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Person Object	N
createdAt	The date and time that the problem was created.	ISO8601 Date String	N
dri	The person who is the DRI for the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Person Object	N
proxy	The person who is designated as the proxy for this problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Person Object	N
epm	The person who is the EPM for the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Person Object	N
originator	The person who created the problem. (See <a href="#">7.2 People Included in Other Objects</a> )	Person Object	N
assigneeLastModifiedAt	The date and time that the problem was modified by the assignee. For a newly assigned problem, this is set to the time of assignment.	ISO8601 Date String	N
duplicateOfProblemID	If the problem has a resolution of "Duplicate", this is the ID of the problem that it was marked as a duplicate of.	Integer	N
reproducible	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String	N
taskOrder	The task order value.	Float	N
counts.relatedProblems	The number of related problems.	Integer	N
isThirdPartyAppRelated	Indications that this problem is related to a third party app.	Boolean	N
dateNeededCurrent	The current value for date needed. (startactualdate)	ISO 8601 date string or null	N
dateNeededOriginal	The original, planned value for date needed. (startplanneddate)	ISO 8601 date string or null	N



Key	Description	Data Type	Default
targetCompletionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date string or null	N
targetCompletionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date string without null	N
targetStartDate	The targeted start date. (duedate)	ISO 8601 date string or null	N
isApproved	Has this work been approved? (featureapproved)	Boolean	N
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean	N
isAutoCalculated	Whether to auto-calculate dates and effort from subtasks.	Boolean	N
hasNewAPIImpact	Does this problem involve new API? (featurenewapi)	Boolean	N
hasNewSPIImpact	Feature has new SPI (featurenewspi)	Boolean	N
hasHumanInterfaceImpact	Does the problem have HI impact? (featurehi)	Boolean	N
hasThirdPartyImpact	Feature has third party content (feature3rdparty)	Boolean	N
hasImportExportImpact	Does this problem require import / export review? (featureimpexp)	Boolean	N
hasLocalizationImpact	Does the problem have localization impact? (featureloc)	Boolean	N
hasPatentReviewImpact	Feature needs patent review (featureneedspatent)	Boolean	N
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean	N
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean	N
effortCurrentTotalEstimate	Effort, current total estimate, in days. (effortcurtotal)	Float or null	N
effortOriginalTotalEstimate	Effort, original total estimate, in days. (effortinittotal)	Float or null	N



Key	Description	Data Type	Default
effortPercentComplete	Effort, percent complete, from 0 to 100. (effortpercentcomplete)	Integer or null	N
effortRemaining	Effort, remaining, in days. (effortremain)	Float or null	N
effortExpended	Effort, expended, in days. (effortexpended)	Float or null	N
testCase	Feature test case (featuretextcaseid)	String or null	N
foundInBuild	The name of the component build that this problem was found in.	String or null	N
fixedInBuild	The name of the component build that this problem was fixed in.	String or null	N
verifiedInBuild	The name of the component build that this problem was verified in.	String or null	N
mustBeFixedInBuild	The name of the component build that this problem must be fixed in.	String or null	N
isVerifiedByTester	Was this problem verified by a tester?	Boolean	N
resolvedAt	The date and time that the issue was resolved.	ISO8601 Date String	N
closedAt	The date and time that the issue was closed.	ISO8601 Date String	N
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Maximum size of 2048 characters.	String or null	N
keywords	An array of keyword objects. (See <a href="#">4.5.1 Get Problem Keywords List</a> )	Keyword meta data for each problem.	N
relatedProblems	An array of related problem Objects	Related Problem Object	N
description	Object representation of description	Description Object	N
diagnosis	Object representation of diagnosis	Diagnosis Object	N
keywordNames	An array containing all the keyword names without its metadata.	Array of keyword names	N



Key	Description	Data Type	Default
event	The problem event object.	Event object or null	N
productSecurity	A Product Security Object	ProductSecurity Object	N
label	Name of the label	String or null	N
scheduledTestIDs	Array of scheduled test IDs added as relations to problem	Array of Integer	N
eventEndDate	Event end date for event of problem	ISO 8601 date and time string or null	N
milestoneEndDate	Milestone end date for milestone of problem	ISO 8601 date and time string or null	N
targetMilestonePlaceholders	Number of placeholders in Target milestone	Integer	N
counts	An object containing counts of problem properties. Counts object is defined in table 1.4.	Object	N

Table 1.1 Product Security Object

Key	Description	Data Type
type	SecurityType code of the product security. ENUMERATED VALUE	String
isPrivacyIssue	Does this product contain a privacy issue.	Boolean or null
cwe	CWE (Not defined). CWE object is defined in table 1.2	CWE Object
color	Color code of the problem. ENUMERATED VALUE	String
vulnerabilityClassification	An object containing vulnerability information. Defined in table 1.3	Vulnerability Object
visibility	Visibility of the issue. ENUMERATED VALUE	String
securityDRI	Person object of the Security DRI	Object or null
securityVerifier	Person object of the Security Verifier	Object or null
securityMasterID	Security Master ID for the product security	Integer



Key	Description	Data Type
foundAt	Date found of the oldest bug in Security umbrella	ISO Date String or null
disclosedAt	Disclosure date of the Product Security	ISO Date String or null
externals	Array containing external names	Array of String

### 1.2 CWE Object

Key	Description	Data Type
primary	List Provided by Product Security Team	String or null
secondary	List Provided by Product Security Team	String or null

### 1.3 Vulnerability Object

Key	Description	Data Type
attackVector	The attack vector. ENUMERATED VALUE	String
authentication	Authentication for the Product Security object. ENUMERATED VALUE	String
assets	Assets code of the Product Security. ENUMERATED VALUE	String
complexity	Complexity of the problem. ENUMERATED VALUE	String
exploitability	Exploitability of the problem. ENUMERATED VALUE	String
impact	Impact of the problem. ENUMERATED VALUE	String
userBase	User base of the problem. ENUMERATED VALUE	String

### 1.4 Counts Object

Key	Description	Data Type
attachments	The number of attachments.	Integer



Key	Description	Data Type
pictures	The number of pictures.	Integer
cc	The number of people who are CCed on the problem.	Integer
relatedProblems	The number of related problems	Integer
targetMilestones	The number of target milestones for a problem	Integer
duplicates	Number of duplicate related problems	Integer
blocking	Number of Blocking related problems	Integer
blockedBy	Number of Blocked-by related problems	Integer
crashes	The problem crash count	Integer
seedUsers	The number of Seed Users	Integer

## [F] Examples

### Simple search by component and state:

Client request:

```
POST /problems/find
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "component": {
    "name": "Radar",
    "version": "6.15"
  },
  "state": "Analyze"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
```



```
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "assignee": {
      "firstName": "Conor",
      "lastName": "Hughes"
    },
    "classification": "Feature (New)",
    "component": {
      "name": "Radar",
      "version": "6.15"
    },
    "fingerprint": "775c96a2",
    "fixOrder": 6,
    "id": 3211968,
    "lastModifiedAt": "2011-11-20T22:56:00+0000",
    "milestone": {
      "component": {
        "name": "Radar",
        "version": "6.15"
      },
      "name": "Later"
    },
    "priority": 1,
    "state": "Analyze",
    "substate": "Screen",
    "title": "not enough monKey in Mac OS X"
  },
  ...
]
```

### Search with IDs Only:

Client request:

```
POST /problems/find
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "component": {
    "name": "Radar",
    "version": "6.15"
  },
  "idsOnly": true,
  "state": "Analyze"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
```





```
Content-Type: application/json; charset=utf-8
[
  3211968,
  3784879,
  ...
]
```

### Search with custom fields requests:

Specifying a list of fields to be returned should work the same as with Section [3.1 Find Problem By ID](#).

Client request:

```
POST /problems/find
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
X-Fields-Requested:
title,component,state,substate,resolution,priority,relatedProblems,keywords
{
  "component": {
    "name": "SUITools",
    "version": "X"
  },
  "state": "Analyze"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "component": {
      "name": "SUITools",
      "version": "X"
    },
    "keywords": [
      {
        "addedAt": "2012-05-17T11:30:00+0000",
        "addedBy": {
          "dsid": 1141936992,
          "email": "rakesh\_k@apple.com",
          "firstName": "Rakesh",
          "lastName": "K",
          "type": "External"
        },
        "keyword": {
          "id": 78790,
          "name": "Radar"
        }
      }
    ]
  }
]
```



```

        }
    ],
    "priority": 1,
    "relatedProblems": [
        ...
    ],
    "resolution": "Unresolved",
    "state": "Analyze",
    "substate": "Screen",
    "title": "not enough monKey in Mac OS X"
},
...
]

```

Optionally, supply the list of fields in the JSON:

```

POST /problems/find
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "component": {
    "name": "Radar",
    "version": "6.15"
  },
  "fieldsRequested": [
    "title",
    "component",
    "state",
    "substate",
    "resolution",
    "priority",
    "relatedProblems"
  ],
  "state": "Analyze"
}

```

Client request:

```

POST /problems/find
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "component": {
    "name": "Radar",
    "version": "6.15"
  },
  "countsOnly": true,
  "state": "Analyze"
}

```

Server response:

```

HTTP/1.1 200 OK

```



```
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "totalCount": 25
}
```

### Search for multiple IDs:

Client request:

```
POST /problems/find
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "id": [
    8000000,
    9000000
  ]
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "id": 8000000, ...
  },
  {
    "id": 9000000, ...
  }
]
```

### Search Problem with Product Security:

Client request:

```
POST /problems/find
X-API-Version: 1.2
Content-Type: application/json; charset=utf-8
{
  "productSecurity": {
    "type": { "gt": "Exposure" },
    "color": { "lt": "Yellow" },
    "isProblemSecurityMaster": true,
    "cwe": { "primary": { "like": "Race Condition" } },
    "reportersCount": { "lt": 5 }
  },
  "countsOnly": true
}
```



Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.2
Content-Type: application/json; charset=utf-8
{
  "totalCount": 300
}
```

Client request:

```
POST /problems/find
X-API-Version: 1.2
Content-Type: application/json; charset=utf-8
{
  "productSecurity": {
    "type": { "eq": "Exposure" },
    "color": { "neq": "Yellow" },
    "isProblemSecurityMaster": false,
    "reporters": { "name": { "like": "Radar" } },
    "reportersCount": { "lt": 5 }
  },
  "countsOnly": true
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.2
Content-Type: application/json; charset=utf-8
{
  "totalCount": 300
}
```

#### **Find Problem Request with non-default security:**

Client request:

```
POST /problems/find
X-API-Version: 1.4
{
  "component": { "name": "Radar", "version": "Automation" },
  "hasDefaultSecurity": false,
  "idsOnly": true
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
```



X-API-Version: 1.4  
Content-Type: application/json; charset=utf-8

[8251807,11135590,13233763,14000523,14002099,14005416,14054042]

### Find Problems based radarWebProgram in originator based search

Client request:

```
POST /problems/find
X-API-Version: 1.4
{
  "originator": {"radarWebProgram": "Radar Training"},
  "idsOnly": true
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.4
Content-Type: application/json; charset=utf-8

[8251807,11135590,13233763,14000523,14002099,14005416,14054042]
```

### Find Problem Request with orderBy condition:

```
POST /problems/find
{
  "component": {
    "name": "SUITools",
    "version": "X"
  }
  "orderBy": [
    { "field": "id", "order": "ascending" },
    { "field": "classification", "order": "ascending" },
    { "field": "priority", "order": "descending" }
  ]
}
```

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "assignee": {
      "firstName": "Conor",
      "lastName": "Hughes"
    },
    "classification": "Feature (New)",
    "component": {
      "name": "SUITools",
      "version": "X"
    }
  }
]
```



```

    },
    "fingerprint": "775c96a2",
    "fixOrder": 6,
    "id": 3211968,
    "lastModifiedAt": "2011-11-20T22:56:00+0000",
    "milestone": {
      "component": {
        "name": "SUITools",
        "version": "X"
      },
      "name": "Later"
    },
    "priority": 1,
    "state": "Analyze",
    "substate": "Screen",
    "title": "not enough monKey in Mac OS X"
  },
  {
    "assignee": {
      "firstName": "Conor",
      "lastName": "Hughes"
    },
    "classification": "Serious Bug",
    "component": {
      "name": "SUITools",
      "version": "X"
    },
    "fingerprint": "787c96n2",
    "fixOrder": 3,
    "id": 3211969,
    "lastModifiedAt": "2011-11-20T22:56:00+0000",
    "milestone": {
      "component": {
        "name": "SUITools",
        "version": "X"
      },
      "name": "Later"
    },
    "priority": 1,
    "state": "Analyze",
    "substate": "Fix",
    "title": "not enough monKey in Mac OS X"
  }
  ...
]

```

#### Find Problem Request with an obscene amount of orderBy condition:

```

POST /problems/find
{
  "component": {
    "name": "SUITools",
    "version": "X"
  }
}

```



```
"orderBy": [
  { "field": "id", "order": "ascending" } ,
  { "field": "classification", "order": "ascending" },
  { "field": "priority", "order": "descending" },
  ...
]
```

```
HTTP/1.1 400 Bad Request
Status: 400
Content-Type: application/json; charset=utf-8
```

```
{
  "status": "400 Bad Request",
  "title": "Too many orderBy conditions specified",
  "message": "There are more orderBy conditions than allowed. Please
try limiting your orderBy conditions to be reasonable.",
  "help": "View documentation at http://radar.apple.com"
}
```

#### Find Problem Request with incorrect orderBy condition:

```
POST /problems/find
{
  "component": {
    "name": "SUITools",
    "version": "X"
  }
  "orderBy": [
    { "field": "id", "order": "GobblyGook" } ,
  ]
}
```

```
HTTP/1.1 400 Bad Request
Status: 400
Content-Type: application/json; charset=utf-8
{
  "status": "400 Bad Request",
  "title": "Unknown orderBy condition",
  "message": "The orderBy conditions can only be enumerated values.",
  "help": "View documentation at http://radar.apple.com"
}
```

#### Find Problem with Other Related Item eq operator:

Client request:

```
POST /problems/find
Content-Type: application/json; charset=UTF-8
{
  "otherRelatedItem":{
    "eq":{
      "system": "Espresso",
```



```
        "url": "<exp2://Ticket/17952942>"
      },
      "idsOnly":true
    }
  }
```

Server response:

```
HTTP/1.1 200 Ok
Status: 200
Content-Type: application/json; charset=UTF-8

[9053901,12412303,12448660,12450115,12468927]
```

**Finding problem whose assignee with dsid 21313124 or 143425. The Radar account for the person with dsid 143425 is no longer active. The search results will only contain matches for dsid 21313124.**

Client request:

```
POST /problems/find
{
  "assignee":{"any":[21313124, 143425]},
  "idsOnly": true
}
```

Server response:

```
HTTP/1.1 200 ok
Content-Type: application/json; charset=utf-8

[1343424, 14134314]
```

**Find Problem request with id and counts.attachments in X-Fields-Requested**

Client request:

```
POST /problems/find
{
  "id": 16156099
}
"X-Fields-Requested:id,counts.attachments"
```

Server response:

```
HTTP/1.1 200 ok
Content-Type: application/json; charset=utf-8

[
  {
    "id": 16156099,
    "counts": {
      "attachments": 21
    }
  }
]
```





```
]
```

### Find Problem request with id and counts in X-Fields-Requested

Client request:

```
POST /problems/find
{
  "id": 16156099
}
"X-Fields-Requested:id,counts"
```

Server response:

```
HTTP/1.1 200 ok
Content-Type: application/json; charset=utf-8
```

```
[
  {
    "id": 16156099,
    "counts": {
      "blockedBy": 0,
      "blocking": 0,
      "duplicates": 0,
      "relatedProblems": 3,
      "pictures": 6,
      "targetMilestones": 1,
      "seedUsers": 0,
      "attachments": 21,
      "crashes": 0,
      "cc": 0
    }
  }
]
```

## 3.3 Create New Problem

### [A] Description

This API is used to create a new problem in radar. The API mimics the UI in that it only allows certain fields to be populated for a new problem; afterward, the client can initiate additional requests to make further modifications. If the problem cannot be created, the server will respond with an appropriate HTTP error code and a hash representing the error. If the problem is created successfully, the server will respond with a JSON representation of the problem as described in [3.1 Get Problem By ID](#). To control the fields that are returned in the response, the client can use the `X-Fields-Requested` header as described in that section.

See [4.3.3 Upload Problem Enclosure](#) for a discussion of uploading enclosures to new problems.

**[B] Schedule**

Required for version 1.0

**[C] URL Scheme**

POST /problems

**[D] Request Parameters**

Key	Description	Data Type	Required
title	Title of the problem. Max Length is 240 Characters	String	Y
component	Component to be added to a problem. See <a href="#">6.5 Components Included in Other Objects</a> .	Object	Y
originator	The DSID of the problem originator. Only users with sufficient privileges can set the originator. Those privileges are granted by membership in a radar group: < <a href="#">rdar://group/14671</a> > CanSetNewProbOriginator (Access); The user executing a new problem command must be in this group in order to set originator to anyone besides himself.	Integer	N
description	Description of the problem	String	Y
diagnosis	Diagnosis of the problem	String	N
classification	Classification of the problem	String	Y
reproducible	Reproducible text for the problem	String	Y
fixOrder	Fix order for the problem	Integer	N
taskOrder	The task order value. Float value can have max of 3 decimal values and 8 digits, but total length should not exceeds 11 characters including radix point.	Float	N
configuration	Configuration of the problem. Max Length is 1000000 characters.	String	N
configurationSummary	Summarized configuration of the problem. Max Length is 240 characters.	String	N
workaround	Workaround of the problem. Max Length is 1000000 characters.	String	N
sourceChanges	Source changes. Max Length is 1000000 characters.	String	N



Key	Description	Data Type	Required
releaseNotes	Release notes. Max Length is 1000000 characters.	String	N
failedModule	The name of the module that failed. Max Length is 60 characters.	String	N
failureDetail	A one-line summary of the failure. Max Length is 60 characters.	String	N
succinctSummaryRootCause	A detailed summary of the failure. Max Length is 2000 characters.	String	N
actionTaken	A description of the corrective action taken. Max Length is 2000 characters.	String	N
dateNeededCurrent	The current value for date needed. (startactualdate)	ISO 8601 date-time string	N
dateNeededOriginal	The original, planned value for date needed. (startplanneddate)	ISO 8601 date-time string	N
targetCompletionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date-time string	N
targetCompletionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date-time string	N
targetStartDate	The targeted start date. (duedate)	ISO 8601 date-time string	N
isApproved	Has this work been approved? (featureapproved)	Boolean	N
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean	N
isAutoCalculated	Whether to auto-calculate dates and effort from subtasks.	Boolean	N
hasNewAPIImpact	Does this problem involve new API? (featurenewapi)	Boolean	N
hasNewSPIImpact	Feature has new SPI (featurenewspi)	Boolean	N
hasHumanInterfaceImpact	Does the problem have HI impact? (featurehi)	Boolean	N
hasThirdPartyImpact	Feature has third party content (feature3rdparty)	Boolean	N
hasImportExportImpact	Does this problem require import / export review? (featureimpexp)	Boolean	N



Key	Description	Data Type	Required
hasLocalizationImpact	Does the problem have localization impact? (featureloc)	Boolean	N
hasPatentReviewImpact	Feature needs patent review (featureneedspatent)	Boolean	N
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean	N
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean	N
effortCurrentTotalEstimate	Effort, current total estimate, in days. (effortcurtotal)	Float	N
effortOriginalTotalEstimate	Effort, original total estimate, in days. (effortinittotal)	Float	N
effortPercentComplete	Effort, percent complete, from 0 to 100. (effortpercentcomplete)	Integer	N
effortRemaining	Effort, remaining, in days. (effortremain)	Float	N
effortExpended	Effort, expended, in days. (effortexpended)	Float	N
testCase	Feature test case (featuretextcaseid). Max Length is 768 characters.	String	N
foundInBuild	The name of the component build that this problem was found in. Max Length is 25 characters.	String	N
fixedInBuild	The name of the component build that this problem was fixed in. Max Length is 25 characters.	String	N
verifiedInBuild	The name of the component build that this problem was verified in. Max Length is 25 characters.	String	N
mustBeFixedInBuild	The name of the component build that this problem must be fixed in. Max Length is 25 characters.	String	N
isVerifiedByTester	Was this problem verified by a tester?	Boolean	N
isRegressionRequired	Must this problem be regressed?	Boolean	N



Key	Description	Data Type	Required
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Max Length is 2048 characters.	String	N

## [E] Examples

### Create new problem:

Client request:

```
POST /problems
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "classification": "Feature (New)",
  "component": {
    "name": "Radar",
    "version": "Automation"
  },
  "description": "Create a new radar problem via POST.",
  "reproducible": "Not Applicable",
  "title": "New radar problem"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "assignee": {
    "dsid": 8794,
    "email": "ewalt@apple.com",
    "firstName": "Alan",
    "lastName": "Ewalt",
    "type": "Employee"
  },
  "assigneeLastModifiedAt": "2011-12-01T18:00:00-0800",
  "attachmentsCount": 0,
  "ccCount": 3,
  "classification": "Feature (New)",
  "component": {
    "name": "Radar",
    "version": "Automation"
  },
  "configurationSummary": null,
  "createdAt": "2011-12-01T18:00:00-0800",
  "duplicateOfProblemID": null,
```



```

    "fingerprint": "2011-12-01T18:00:00.000000-08:00",
    "fixOrder": 6,
    "hasReleaseNotes": false,
    "hasSourceChanges": false,
    "hasWorkaround": false,
    "id": 10515963,
    "isReadByAssignee": false,
    "keywordsCount": 0,
    "lastModifiedAt": "2011-12-01T18:00:00-0800",
    "milestone": null,
    "originator": {
      "dsid": 107254372,
      "email": "joel.young@apple.com",
      "firstName": "Joel",
      "lastName": "Young",
      "type": "Employee"
    },
    "otherRelatedItemsCount": 0,
    "picturesCount": 0,
    "priority": 5,
    "relatedProblemsCount": 0,
    "reproducible": "Not Applicable",
    "resolution": "Unresolved",
    "securityListCount": 0,
    "state": "Analyze",
    "substate": "Screen",
    "targetMilestonesCount": 0,
    "thirdPartyProductsCount": 0,
    "title": "New radar problem"
  }
}

```

### 3.4 Update Problem

#### [A] Description

This API is used to update one or more fields in a problem. The POST body should contain a JSON object containing only the Key that should be updated. Any fields that are not represented in the JSON object should not be modified. The server will respond with 200 `Success` and no response body.

**NOTE:** The fingerprint attribute **is optional** in the request body. If the fingerprint is passed and doesn't match the server's copy, it will respond with 409 `Conflict`.

Problem collections, such as `relatedProblems` or `keywords`, can be modified along with the problem, with behavior identical to the sections described in [4 Problem Collections](#). The exception to this is enclosures: since multiple request bodies are not allowed, enclosures must be uploaded as separate requests (see [4.3 Problem Enclosures](#)). In general, collections specified in an update request will be APPENDED to the current value of the collection.

The update of one or more fields in a problem can only be done if the logged in user have access to update that field. Otherwise, the server will respond with 403 `Forbidden`.



The Update Problem API is modified to include one more attribute, seedUsers. User can update the seedUsers value similar to existing crash count attribute.

This count will be updated only if the person calling the API is part of “Radar Can Update Seed Count” radar access group

#### [B] URL Scheme

POST /problems/<problem\_id>

#### [C] Request Attributes

Key	Description	Data Type
title	Title of the problem. Max Length is 240 characters.	String
component	The component that the problem belongs to. See <a href="#">6.5 Components Included in Other Objects</a> .	Object
originator	The DSID of the person who created the problem. Only users with sufficient privileges can set the originator.	Integer
assignee	The DSID of the person who is assigned to the problem.	Integer
dri	The DSID of the person who is the DRI for the problem. Specify null to clear the DRI.	Integer or null
resolvedBy	The DSID of the person who resolved the problem. Specify null to clear the resolver.	Integer or null
state	The problem's state. Valid values are described by <a href="#">10.1 Get Field Enumeration</a> . Changing the state may require changes to the milestone, priority and resolution fields.	String
substate	The problem's substate. This can only be supplied when state is “Analyze”. Valid values are described by <a href="#">10.1 Get Field Enumeration</a> . Changing the substate may require changes to the milestone and priority fields.	String
resolution	The problem's resolution. Valid values are described by <a href="#">10.1 Get Field Enumeration</a> . If set to "Duplicate", the duplicateOfProblemID field must also be supplied.	String



Key	Description	Data Type
isReadByAssignee	Is the problem read by assignee.	Boolean
duplicateOfProblemID	ID of the problem for which current problem is duplicate.	Integer
classification	The problem's classification. Valid values are described by <a href="#">10.1 Get Field Enumeration</a> .	String
reproducible	The problem's reproducibility. Valid values are described by <a href="#">10.1 Get Field Enumeration</a> .	String
milestone	A milestone string for the component. Use 6.10 Get Component Milestone List to get the list of milestones. Max Length is 25 characters.	String
priority	The priority value	Integer
fixOrder	The fix order value	Integer
taskOrder	Task Order of a problem	Float or null
diagnosis	Problem Diagnosis to be appended	String
description	Problem Description to be appended	String
configuration	The full configuration information. Max Length is 1000000 characters.	String or null
configurationSummary	The one-line configuration summary. Max Length is 240 characters.	String or null
sourceChanges	The problem's source changes. Max Length is 1000000 characters.	String or null
keywords	An array of keyword strings or ids to be added to the problem.	Array of strings or integers
relatedProblems	An array of problem relation objects. (See <a href="#">4.4.2 Set Related Problems List</a> )	Array of objects
otherRelatedItems	An array of "other related items" to be added to the problem. (See <a href="#">4.6.3 Set List of Related Items in External Systems</a> )	Array of objects
workaround	A description of the workaround. Max Length is 1000000 characters.	String





Key	Description	Data Type
thirdPartyProducts	An array of “3rd party products” information to be added to the problem. (See <a href="#">4.7.2 Set Third Party Product List</a> )	Array of objects
releaseNotes	The full release notes. Max Length is 1000000 characters.	String
securityList	An array of person or group objects to be added to the problem. (See <a href="#">4.8.3 Append to Security List</a> )	Array of objects
dateNeededCurrent	The current value for date needed. (startactualdate)	ISO 8601 date string or null
dateNeededOriginal	The original, planned value for date needed. (startplanneddate)	ISO 8601 date string or null
targetCompletionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date string or null
targetCompletionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date string or null
targetStartDate	The targeted start date. (duedate)	ISO 8601 date string or null
isApproved	Has this work been approved? (featureapproved)	Boolean
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean
hasNewAPIImpact	Does this problem involve new API? (featurenewapi)	Boolean
isAutoCalculated	Is effort ,dateNeeded and Target Completion should be auto calculated.	Boolean
hasThirdPartyImpact	Feature has third party content (feature3rdparty)	Boolean
hasPatentReviewImpact	Feature needs patent review (featureneedspatent)	Boolean
hasNewSPIImpact	Feature has new SPI (featurenewspi)	Boolean
hasImportExportImpact	Does this problem require import / export review? (featureimpexp)	Boolean
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean
hasHumanInterfaceImpact	Does the problem have Human Interface impact? (featurehi)	Boolean



Key	Description	Data Type
hasLocalizationImpact	Does the problem have localization impact? (featureloc)	Boolean
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean
effortCurrentTotalEstimate	Effort, current total estimate (effortcurtotal)	Float or null
effortOriginalTotalEstimate	Effort, original total estimate (effortinittotal)	Float or null
effortPercentComplete	Effort, percent complete (effortpercentcomplete)	Integer or null
effortRemaining	Effort, remaining (effortremain)	Float or null
effortExpended	Effort, expended (effortexpended)	Float or null
testCase	Feature test case (featuretextcaseid). Max Length is 768 characters.	String or null
foundInBuild	The name of the component build that this problem was found in. Max Length is 25 characters.	String or null
fixedInBuild	The name of the component build that this problem was fixed in. Max Length is 25 characters.	String or null
verifiedInBuild	The name of the component build that this problem was verified in. Max Length is 25 characters.	String or null
mustBeFixedInBuild	The name of the component build that this problem must be fixed in. Max Length is 25 characters.	String or null
isVerifiedByTester	Was this problem verified by a tester?	Boolean
isRegressionRequired	Must this problem be regressed?	Boolean
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Max Length is 2048 characters.	String or null
failedModule	The name of the module that failed. Max Length is 60 characters.	String or null
failureDetail	A one-line summary of the failure. Max Length is 60 characters.	String or null
succinctSummaryRootCause	A detailed summary of the failure. Max Length is 2000 characters.	String or null



Key	Description	Data Type
actionTaken	A description of the corrective action taken. Max Length is 2000 characters.	String or null
event	A event string for the component. Max Length is 25 characters.	String or null
productSecurity	A productSecurity object	ProductSecurity Object
labelID	LabelID of the label. To remove label from problem 'null' should be used.	Integer or null
isThirdPartyAppRelated	Does this problem has ThirdParty Product added?	Boolean
counts	An object containing counts of problem properties. Counts object is defined in table 1.4.	Object

Table 1.1 Product Security Object

Key	Description	Data Type
type	SecurityType of the product security. ENUMERATED VALUE	String
isPrivacyIssue	Does this product contain a privacy issue.	Boolean or null
cwe	CWE (Not defined). CWE object is defined in table 1.2	CWE Object
color	Color code of the problem. ENUNMERATED VALUE	String
vulnerabilityClassification	An object containing vulnerability information. Defined in table 1.3	Vulnerability Object
visibility	Visibility of the issue. ENUMERATED VALUE	String
securityDRI	DSID of security DRI	Integer or null
securityVerifier	DSID of security Verifier	Integer or null
securityMasterID	Security Master ID for the product security	Integer
foundAt	Date found of the oldest bug in Security umbrella	ISO8601 Date String
disclosedAt	Disclosure date of the Product Security	ISO8601 Date String

## 1.2 CWE Object



Key	Description	Data Type
primary	List Provided by Product Security Team	String or null
secondary	List Provided by Product Security Team	String or null

### 1.3 Vulnerability Object

Key	Description	Data Type
attackVector	The attack vector. ENUMERATED VALUE	String
authentication	Authentication for the Product Security object. ENUMERATED VALUE	String
assets	Assets code of the Product Security. ENUMERATED VALUE	String
complexity	Complexity of the problem. ENUMERATED VALUE	String
exploitability	Exploitability of the problem. ENUMERATED VALUE	String
impact	Impact of the problem. ENUMERATED VALUE	String
userBase	User base of the problem. ENUMERATED VALUE	String

### 1.4 Counts Object

Key	Description	Data Type
seedUsers	Number of seed users encountering this issue. <b>This value is only editable by accounts held by the Apple Seed program.</b>	String

### [E] Examples

#### Modify Problem with an out-of-date fingerprint:

Client request:

```
POST /problems/9000000
X-API-Version: 1.0
{
  "fingerprint": "old value",
  "state": "Integrate",
  ...
}
```



Server response:

```
HTTP/1.1 409 Conflict
Status: 409
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "help": "Please contact radar-help@group.apple.com",
  "message": "The radar you are attempting to modify has already been
modified by someone else. Please reload the radar before attempting to
save.",
  "status": "409 Conflict",
  "title": "Conflict"
}
```

### Modify the Title and Diagnosis:

Note that modified fields will not necessarily be returned in the response.

Client request:

```
POST /problems/9000000
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "diagnosis": "Since a foldable MacBook Air would indeed be magic,
the title should reflect it.",
  "fingerprint": "...",
  "title": "MAGIC Foldable Mac Book Air"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
```

### Modify the Radar State:

Client request:

```
POST /problems/9000000
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "fingerprint": "...",
  "state": "Integrate",
  "resolution": "Software Changed",
}
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
```



Status: 200

### Modify the Radar Product Security:

Client request:

```
POST /problems/9000000
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "productSecurity":
  {
    "type": "Exposure",
    "isPrivacyIssue": false,
    "foundAt": "2012-12-11",
    "securityDRI": 1118580968,
    "cwe":{"primary": "Race Condition",
    "secondary": null},
    "vulnerabilityClassification":{"
    "attackVector": "Local System",
    "authentication": "Authenticated",
    "assets": "General"
    }
  }
}
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.2
Status: 200
```

## 3.5 Modify Multiple Problems

### [A] Description

This API is used to update one or more fields of a set of problem. The POST body should contain a JSON object containing only the keys that should be updated and the list of Problem IDs. Any fields that are not represented in the JSON object should not be modified. On success, the server will respond with 200 OK.

If any one of the fields cannot be modified for a given radar, then all fields are not modified for that radar. In case of any of radar is failed to update then the error message with that problem Id shall be displayed in the response body.

Below Parameters -

'clearFoundInBuild','clearFixedInBuild','clearMustBeFixedByBuild','clearVerifiedInBuild','clearMilestone','clearDRI','clearResolver','clearSerialNo','clearEvent','clearEPM','clearSecurityDRI','clearSecurityVerifier'.



are mutually exclusive fields and only one of them can be updated at a time. An error message will be thrown when more than one mutually exclusive fields are passed in request.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

POST /problems/modify

#### [D] Request Attributes

Key	Description	Data Type
problemIDs	List of the problem IDs to be modified.	Array of Integer
component	The component that the problem belongs to. See <a href="#">1.0 Components Attributes</a> .	Object
originator	The DSID of the person who created the problem. Only users with sufficient privileges can set the originator.	Integer
assignee	The DSID of the person who is assigned to the problem.	Integer
dri	The DSID of the person who is the DRI for the problem.	Integer
resolvedBy	The DSID of the person who resolved the problem.	Integer
epm	The EPM of the problem	Integer
state	The problem's state. Changing the state may require changes to the milestone, priority and resolution fields.	String
substate	The problem's substate. This can only be supplied when state is "Analyze". Changing the substate may require changes to the milestone and priority fields.	String
resolution	The problem's resolution. If set to "Duplicate", the duplicateOfProblemID field must also be supplied.	String



Key	Description	Data Type
duplicateOfProblemID	ID of the problem for which current problem is duplicate.	Integer
classification	The problem's classification.	String
reproducible	The problem's reproducibility.	String
milestone	The problem milestone. (See <a href="#">6.3.3 Component Milestones Included in Other Objects</a> ). If milestone object is included than component object should also be included in request and component object should contain same value as mentioned inside milestone object.	Object
priority	The priority value	Integer
event	The event of the problem	String
fixOrder	The fix order value	Integer
diagnosis	Problem Diagnosis to be appended	String
addKeywordById	ID of the keyword to be added.	String
removeKeywordById	ID of the keyword to be removed.	String
addSecurity	Person/Group to be added.See 1.1 Security Attributes for addSecurity Object.	Object
removeSecurity	Person/Group to be added.See 1.1 Security Attributes for addSecurity Object.	Object
addCC	DSId of person to be added to CC List.	Integer
removeCC	DSId of person to be removed from CC List	Integer
targetMilestone	A Targetmilestone Object to be modify.(See 1.1 Targetmilestone attributes.)	Object
adcNotes	For adding text to ADC notes	String
isReadByAssignee	A Boolean value for setting read by assignee field	Boolean
moveStateForward	A Boolean value for moving State forward.	Boolean
moveStateBack	A Boolean value for moving State back.	Boolean
clearFoundInBuild	A Boolean value to clear found in build	Boolean





Key	Description	Data Type
clearMustBeFixedByBuild	A Boolean value to clear must be fixed by build	Boolean
clearFixedInBuild	A Boolean value to clear fixed in build	Boolean
clearVerifiedInBuild	A Boolean value to clear verified in build	Boolean
clearMilestone	A Boolean value to clear Milestone	Boolean
clearDRI	A Boolean value to clear DRI	Boolean
clearResolver	A Boolean value to clear Resolver	Boolean
clearSerialNo	A Boolean value to clear Serial number	Boolean
clearEvent	A Boolean value to clear Event	Boolean
clearEPM	A Boolean value to clear EPM	Boolean
clearSecurityDRI	A Boolean value to clear Security DRI	Boolean
clearSecurityVerifier	A Boolean value to clear Security Verifier	Boolean
relatedProblems	A problem relation objects. (See 1.2 for related problem attributes.)	Object
dateNeededCurrent	The current value for date needed. (startactualdate)	ISO 8601 date string
dateNeededOriginal	The original, planned value for date needed. (startplanneddate)	ISO 8601 date string
targetCompletionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date string
targetCompletionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date string
targetStartDate	The targeted start date. (duedate)	ISO 8601 date string
isApproved	Has this work been approved? (featureapproved)	Boolean
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean
isAutoCalculated	Is Auto-calculate Dates and Effort from subtasks?	Boolean



Key	Description	Data Type
hasNewAPIImpact	Does this problem involve new API? (featurenewapi)	Boolean
hasThirdPartyImpact	Feature has third party content (feature3rdparty)	Boolean
hasPatentReviewImpact	Feature needs patent review (featureneedspatent)	Boolean
hasNewSPIImpact	Feature has new SPI (featurenewspi)	Boolean
hasImportExportImpact	Does this problem require import / export review? (featureimpexp)	Boolean
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean
hasHumanInterfaceImpact	Does the problem have HI impact? (featurehi)	Boolean
hasLocalizationImpact	Does the problem have localization impact? (featureloc)	Boolean
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean
effortCurrentTotal	Effort, current total estimate (effortcurtotal)	Float
effortOriginalTotal	Effort, original total estimate (effortinittotal)	Float
effortPercentComplete	Effort, percent complete (effortpercentcomplete)	Integer
effortRemaining	Effort, remaining (effortremain)	Float
effortExpended	Effort, expended (effortexpended)	Float
taskOrder	Task Order of a problem	Float or null
testCase	Feature test case (featuretextcaseid)	String
foundInBuild	The name of the component build that this problem was found in.	String
fixedInBuild	The name of the component build that this problem was fixed in.	String
verifiedInBuild	The name of the component build that this problem was verified in.	String



Key	Description	Data Type
fixedByInBuild	The name of the component build that this problem fixed by.	String
isVerifiedByTester	Was this problem verified by a tester?	Boolean
type	Product Security Type	String
verifiedByPS	Verified by Product Security	String
affectedProductID	Affected product ID	Integer
isPrivacyIssue	Does this product contain a privacy issue.	Boolean or null
cwe	CWE (Not defined). CWE object is defined in table 1.3	CWE Object
color	Color code of Product Security. Setting color will make the isOverride attribute as true.	String
attackVector	Attack vector code of the Product Security	String
authentication	Authentication code of the Product Security	String
assets	Assets code of the Product Security	String
complexity	Complexity code of the Product Security	String
exploitability	Attack vector code of the Product Security	String
impact	Impact code of the Product Security	String
userBase	User Base code of the Product Security	String
visibility	Visibility code of the Product Security	String
securityDRI	DSID of the Security DRI	Integer
securityVerifier	DSID of the Security Verifier	Integer
changeFromRecomReleaseVehicleId	Id of the recommended release vehicle to be changed	Integer
changeToRecomReleaseVehicleId	Id of the recommended release vehicle to be changed to.	Integer
affectedProductID	Id of the affected product to be added.	Integer
changeFromPlanReleaseVehicleId	Id of the planned release vehicle to be changed	Integer



Key	Description	Data Type
changeToPlanReleaseVehicleId	Id of the planned release vehicle to be changed to.	Integer
buildInfo	Build info of problem	String
labelID	The id of the label	Integer

**Table 1.0. Attributes for component field.**

Key	Description	Data Type
name	Name of the component.	String
version	Version of the component	String

**Table 1.1. Attributes for milestone field.**

Key	Description	Data Type
oldMilestoneName	Name of the milestone to be changed.	String
componentName	Name of the component	String
componentVersion	Version of the component	String
newMilestoneName	Name of the new milestone.	String

**Table 1.2. Attributes for related problem field**

Key	Description	Data Type
type	One of: "related-to", "blocked-by", "blocking", "parent-of", "subtask-of". Other relation types (such as "duplicate-of", "clone-of") are not allowed.	String
problemID	Related problem ID	Integer

### 1.3 CWE Object

Key	Description	Data Type
primary	List Provided by Product Security Team	String
secondary	List Provided by Product Security Team	String

**[E] Examples****(i) Update multiple problems**

Client request:

```
POST /problems/modify
X-API-Version: 1.3
Content-Type: application/json
{
  "problemIDs": [
    12309822,
    12309824,
    12309825,
    12309828
  ],
  "component": {
    "name": "Radar",
    "version": "7.1"
  },
  "state": "Analyze",
  "priority": 3
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
X-API-Version: 1.3
```

**(ii) Update multiple problems (Response with Error Message)**

Client request:

```
POST /problems/modify
X-API-Version: 1.3
Content-Type: application/json
{
  "problemIDs": [
    12309881,
    12309874,
  ],
  "state": "Integrate"
}
```

Server response:

```
HTTP/1.1 400 Bad Request
Status: 400
Content-Type: application/json; charset=utf-8
X-API-Version: 1.3
[
  {
    "problemID": 12309874,
```



```

    "errorMsg": "Resolution cannot be set to 'Unresolved' if the
state is greater than 'Analyze'.",
    "problemTitle": "testing"
  }
]

```

### 3.6 Clone Problem

#### [A] Description

This API is used to clone problem in radar. The API mimics the UI in that it only allows certain fields to be populated for a cloned problem; afterward, the client can initiate additional requests to make further modifications. If the problem cannot be cloned, the server will respond with an appropriate HTTP error code and a hash representing the error. If the problem is cloned successfully, the server will respond with a JSON representation of the problem as described in [3.1 Get Problem By ID](#). To control the fields that are returned in the response, the client can use the `X-Fields-Requested` header as described in that section, or the `fieldsRequested` option in the post body.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

POST /problems/<problem\_id>/clone

#### [D] Request Parameters

Key	Description	Data Type	Required
originatorOfClone	Originator of cloned problem. If set as true then current user will be set as originator otherwise original problem originator will be set as originator	Boolean	Y
component	Component for the new cloned problem. See <a href="#">6.5 Components Included in Other Objects</a> .	Object	Y
reason	The reason for cloning the problem. Max length 1100 characters.	String	Y
title	The title of the new cloned problem.	String	N
classification	The classification of the new cloned problem	String	N
fixOrder	The fixOrder of the new cloned problem	Integer	N
reproducible	The reproducibility of the new cloned problem	String	N



## [E] Response Parameters

Response Attribute of Cloned Problem will be same as Response Attribute of Sec 3.1 GetProblemDetails API.

## [F] Example

### Creating a Clone problem from existing problem:

Client request:

```
POST /problems/14648918/clone
Content-Type: application/json; charset=utf-8
{
  "component": {
    "name": "Radar",
    "version": "7.2"
  },
  "originatorOfClone": true,
  "reason": "Cloning for RWS"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
Content-Type: application/json; charset=utf-8
{
  "hasSourceChanges": false,
  "reproducible": "I Didn't Try",
  "state": "Analyze",
  "resolution": "Unresolved",
  "id": 14648934,
  "milestone": {
    "component": {
      "name": "Radar",
      "version": "7.2"
    },
    "name": "Later"
  },
  "fingerprint": "5f38548e",
  "title": "Foldable Mac Book Air",
  "isExternallyViewable": false,
  "component": {
    "name": "Radar",
    "version": "7.2"
  },
  "createdAt": "2011-02-14T21:50:17+0000",
  "priority": 2,
  "lastModifiedAt": "2014-01-17T07:26:19+0000",
  "hasReleaseNotes": false,
  "isReadByAssignee": false,
  "fixOrder": 6,
```



```

"assignee": {
  "lastName": "Uchida",
  "email": "xxxx@apple.com",
  "dsid": 1433115649,
  "firstName": "Nicholas",
  "type": "Employee"
},
"assigneeLastModifiedAt": "2014-01-17T07:26:19+0000",
"substate": "Screen",
"duplicateOfProblemID": null,
"configurationSummary": null,
"classification": "Task",
"hasWorkaround": false,
"counts": {
  "securityList": 1,
  "duplicates": 0,
  "keywords": 5,
  "pictures": 3,
  "seedUsers": 0,
  "targetMilestones": 0,
  "relatedTests": 0,
  "crashes": 0,
  "blockedBy": 0,
  "blocking": 0,
  "thirdPartyProducts": 0,
  "relatedProblems": 18,
  "otherRelatedItems": 0,
  "attachments": 0,
  "cc": 92
},
"originator": {
  "lastName": "Vaithyanathan",
  "email": "xxxx@apple.com",
  "dsid": 187179976,
  "firstName": "Kartik",
  "type": "Employee"
}
}

```

### 3.7 Create New Problem With Attachments and Pictures

#### [A] Description

This API will be used to create a new Radar problem as well as attach files and/or images simultaneously. The existing problems/ route will be used;

- if the Content-Type is application/json, the semantics of the existing API will be used.
- if the Content-Type is multipart with a subtype of related, the semantics detailed below will be observed.

A whitelist component check will be done for developer user. If developer user tries to pass any other component which are not mentioned below then an error message will be shown.



**[B] Schedule**

Required for version 1.5

**[C] URL Scheme**

POST /problems

problemFields needs to be posted as a form body and on uploading the file, request will be treated as multipart.

**[D] Request Parameters**

Key	Description	Data Type	Required
title	Title of the problem. Max Length is 240 Characters	String	Y
component	Component to be added to a problem. Object should contain component name and version or componentID	Object or Integer	Y
originator	The DSID of the problem originator. Only users with sufficient privileges can set the originator. Those privileges are granted by membership in a radar group: <a href="rdar://group/14671">&lt;rdar://group/14671&gt;</a> CanSetNewProbOriginator (Access); The user executing a new problem command must be in this group in order to set originator to anyone besides himself.	Integer	N
description	Description of the problem	String	Y
diagnosis	Diagnosis of the problem	String	N
classification	Classification of the problem	String	Y
reproducible	Reproducible text for the problem	String	Y
fixOrder	Fix order for the problem	Integer	N
taskOrder	The task order value. Float value can have max of 3 decimal values and 8 digits, but total length should not exceeds 11 characters including radix point.	Float	N
configuration	Configuration of the problem. Max Length is 1000000 characters.	String	N
configurationSummary	Summarized configuration of the problem. Max Length is 240 characters.	String	N
workaround	Workaround of the problem. Max Length is 1000000 characters.	String	N



Key	Description	Data Type	Required
sourceChanges	Source changes. Max Length is 1000000 characters.	String	N
releaseNotes	Release notes. Max Length is 1000000 characters.	String	N
failedModule	The name of the module that failed. Max Length is 60 characters.	String	N
failureDetail	A one-line summary of the failure. Max Length is 60 characters.	String	N
succinctSummaryRootCause	A detailed summary of the failure. Max Length is 2000 characters.	String	N
actionTaken	A description of the corrective action taken. Max Length is 2000 characters.	String	N
dateNeededCurrent	The current value for date needed. (startactualdate)	ISO 8601 date-time string	N
dateNeededOriginal	The original, planned value for date needed. (startplanneddate)	ISO 8601 date-time string	N
targetCompletionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date-time string	N
targetCompletionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date-time string	N
targetStartDate	The targeted start date. (duedate)	ISO 8601 date-time string	N
isApproved	Has this work been approved? (featureapproved)	Boolean	N
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean	N
isAutoCalculated	Whether to auto-calculate dates and effort from subtasks.	Boolean	N
hasNewAPIImpact	Does this problem involve new API? (featurenewapi)	Boolean	N
hasNewSPIImpact	Feature has new SPI (featurenewspi)	Boolean	N
hasHumanInterfaceImpact	Does the problem have HI impact? (featurehi)	Boolean	N
hasThirdPartyImpact	Feature has third party content (feature3rdparty)	Boolean	N



Key	Description	Data Type	Required
hasImportExportImpact	Does this problem require import / export review? (featureimpexp)	Boolean	N
hasLocalizationImpact	Does the problem have localization impact? (featureloc)	Boolean	N
hasPatentReviewImpact	Feature needs patent review (featureneedspatent)	Boolean	N
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean	N
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean	N
effortCurrentTotalEstimate	Effort, current total estimate, in days. (effortcurtotal)	Float	N
effortOriginalTotalEstimate	Effort, original total estimate, in days. (effortinittotal)	Float	N
effortPercentComplete	Effort, percent complete, from 0 to 100. (effortpercentcomplete)	Integer	N
effortRemaining	Effort, remaining, in days. (effortremain)	Float	N
effortExpended	Effort, expended, in days. (effortexpended)	Float	N
testCase	Feature test case (featuretextcaseid). Max Length is 768 characters.	String	N
foundInBuild	The name of the component build that this problem was found in. Max Length is 25 characters.	String	N
fixedInBuild	The name of the component build that this problem was fixed in. Max Length is 25 characters.	String	N
verifiedInBuild	The name of the component build that this problem was verified in. Max Length is 25 characters.	String	N
mustBeFixedInBuild	The name of the component build that this problem must be fixed in. Max Length is 25 characters.	String	N



Key	Description	Data Type	Required
isVerifiedByTester	Was this problem verified by a tester?	Boolean	N
isRegressionRequired	Must this problem be regressed?	Boolean	N
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Max Length is 2048 characters.	String	N

**[E] MIME Parts**

Parameter		Description	Data Type	Required
Problem Data	Content-Type	Must be application/json	String	Y
	Content-Disposition	Must be post-body	String	Y
		MIME data for the Problem JSON Object follows the headers.	Object	Y
File Data	Content-Type	MIME type for attached file. (i.e. application/zip or image/jpeg)	String	N
	Content-Disposition	Must be file-data.	Binary	N
	filename	Filename to be used when storing the attachment.	String	N
	size	Size in bytes of the file to be stored.	String	N
	picture	Indicates that the file should be stored as a picture and is not required to be explicitly downloaded in Radar.	Boolean	N



### [E] Request parameters Supported for Developer User

Key	Description	Data Type	Required
title	Title of the problem. Max Length is 240 Characters	String	Y
component	Component to be added to a problem. Object should contain component name and version or ComponentID.	Object or Integer	Y
description	Description of the problem	String	Y
classification	Classification of the problem	String	Y
reproducible	Reproducible text for the problem	String	Y

### [G] Request Parameters

Create a new problem with attachment

Client request:

```

POST /problems
Content-Type: multipart/related; boundary="__the_boundary__"

--__the_boundary__
Content-Disposition: form-data
Content-Type: application/json; charset=utf-8
{
  "classification": "Feature (New)",
  "component": {
    "name": "Radar",
    "version": "Automation"
  },
  "description": "Create a new radar problem via POST.",
  "reproducible": "Not Applicable",
  "title": "New radar problem"
  "keywords": [
    "Radar BRB",
    "Radar TSTT BRB"
  ]
}
--__the_boundary__
Content-Disposition: file-data; filename="Test_or_ID.zip"; size=1515;
picture=false
Content-Type: application/zip

<.....binary data.....>

--__the_boundary__
Content-Disposition: file-data; filename="test.jpg"; size=145851;
picture=true

```



Content-Type: image/jpeg

<.....binary data.....>

--\_\_the\_boundary\_\_--

Server response:

```
HTTP/1.1 201 Created
Status: 201
Content-Type: application/json; charset=utf-8
{
  "assignee": {
    "dsid": 8794,
    "email": "ewalt@apple.com",
    "firstName": "Alan",
    "lastName": "Ewalt",
    "type": "Employee"
  },
  "assigneeLastModifiedAt": "2011-12-01T18:00:00-0800",
  "attachmentsCount": 0,
  "ccCount": 3,
  "classification": "Feature (New)",
  "component": {
    "name": "Radar",
    "version": "Automation"
  },
  "configurationSummary": null,
  "createdAt": "2011-12-01T18:00:00-0800",
  "duplicateOfProblemID": null,
  "fingerprint": "2011-12-01T18:00:00.000000-08:00",
  "fixOrder": 6,
  "hasReleaseNotes": false,
  "hasSourceChanges": false,
  "hasWorkaround": false,
  "id": 10515963,
  "isReadByAssignee": false,
  "keywordsCount": 0,
  "lastModifiedAt": "2011-12-01T18:00:00-0800",
  "milestone": null,
  "originator": {
    "dsid": 107254372,
    "email": "joel.young@apple.com",
    "firstName": "Joel",
    "lastName": "Young",
    "type": "Employee"
  },
  "otherRelatedItemsCount": 0,
  "picturesCount": 0,
  "priority": 5,
  "relatedProblemsCount": 0,
  "reproducible": "Not Applicable",
  "resolution": "Unresolved",
```



```

    "securityListCount": 0,
    "state": "Analyze",
    "substate": "Screen",
    "targetMilestonesCount": 0,
    "thirdPartyProductsCount": 0,
    "title": "New radar problem"
  }

```

Note : There will be a limit of 10MB on attaching the pictures. If the picture size exceeds 10MB then it will be uploaded to Kona.

#### DEVELOPER EXAMPLES:

```

POST /problems
Content-Type: multipart/related; boundary="__the_boundary__"

--__the_boundary__
Content-Disposition: form-data
Content-Type: application/json; charset=utf-8
{
  "classification": "Feature (New)",
  "component": {
    "name": "iPhone",
    "version": "( new bugs )"
  },
  "description": "Create a new radar problem via POST.",
  "reproducible": "Not Applicable",
  "title": "New radar problem"
  "keywords": [
    "Radar BRB",
    "Radar TSTT BRB"
  ]
}

--__the_boundary__
Content-Disposition: file-data; filename="test.jpg"; size=145851;
picture=true
Content-Type: image/jpeg

<.....binary data.....>

--__the_boundary__--

```

Server response:

```

HTTP/1.1 200 Ok
Status: 200
Content-Type: application/json; charset=utf-8
{
  "id": 10515963,
  "title": "New radar problem",
}

```



### Non-WhiteList Component problem creation by Developer

```
POST /problems
Content-Type: multipart/related; boundary="__the_boundary__"

--__the_boundary__
Content-Disposition: form-data
Content-Type: application/json; charset=utf-8
{
  "classification": "Feature (New)",
  "component": {
    "name": "Radar",
    "version": "Automation"
  },
  "description": "Create a new radar problem via POST.",
  "reproducible": "Not Applicable",
  "title": "New radar problem"
  "keywords": [
    "Radar BRB",
    "Radar TSTT BRB"
  ]
}

--__the_boundary__
Content-Disposition: file-data; filename="test.jpg"; size=145851;
picture=true
Content-Type: image/jpeg

<.....binary data.....>

--__the_boundary__--
```

Server response:

```
HTTP/1.1 403 Forbidden
Status: 403
Content-Type: application/json; charset=utf-8
{
  "status": "403Forbidden",
  "title": "ProblemCreationForbidden",
  "message": "You do not have enough privilege to submit a new bug to
given component.",
  "help": "View documentation at http: //radar.apple.com"
}
```

## 3.8 Get Problem Protection Mask

### [A] Description

This API will be used to return an object that describes the field-level permissions for a given Radar Problem.





The object contains all fields requested as the keys with a enumerated value of 'modifiable' or 'read-only'.

#### [B] Schedule

Required for version 1.5

#### [C] URL Scheme

```
GET problems/<problem_id>/mask
```

#### [D] Examples

##### Get Protection Mask for a Problem:

Client request:

```
GET problems/9000000/mask
X-Fields-Requested:
title,classification,priority,reproducible,id,securityList,component,assignee
```

Server response:

```
HTTP/1.1 200 Ok
Status: 200
Content-Type: application/json; charset=UTF-8
{
  "title": "modifiable",
  "classification": "modifiable",
  "priority": "modifiable",
  "reproducible": "modifiable",
  "id": "read-only",
  "securityList": "read-only",
  "component": "read-only",
  "assignee": "read-only"
}
```

### 3.9 Problems Included in Other Objects

#### [A] Description

Except where specified otherwise, a problem included in another object (such as in 4.4.1 Get Related Problems List) will include the attributes specified here.

#### [B] Schedule

Required for version 1.0

**[C] Attributes**

Key	Description	Data Type
id	The problem ID.	Integer
assignee	The person who the problem is assigned to. See <a href="#">7.2 People Included in Other Objects</a> .	Object
component	The problem's component. See <a href="#">6.5 Components Included in Other Objects</a> .	Object
state	An enumerated string value. Possible values can be fetched using <a href="#">10.1 Get Field Enumeration</a> .	String
title	The problem's title.	String



## 4. PROBLEM COLLECTIONS

For radar attributes that consist of variable-length collections of other records, such as `description` and `relatedProblems`, these collections can be accessed via their own routes, defined below. The description of response formats for these collections also apply to when the collections are requested as part of a problem request, as in [3.1 Get Problem By ID](#) or [3.2 Find Problems](#).

In general, the URL path for a multi-word collection will be a "dasherized" version of the JSON property appended to the URL path for the problem that it belongs to. E.g.: the `relatedProblems` collection for Radar 9000000 uses the path `/problems/9000000/related-problems`.

### 4.1 Problem Description

This group of APIs deals with fetching and appending to the problem description array.

#### 4.1.1 Get Problem Description

##### [A] Description

This API is used to get/read the list of description comments with timestamp and authors.

If the user wants to fetch description comments as plain text, then 'Accept' key with value 'text/plain' needs to be passed as a header. If header is not passed then this API will fetch description as array.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
GET /problems/<problem_id>/description
```

##### [D] Response Attributes

Key	Description	Data Type
addedBy	The person who created the entry. This object will only include the full name and email address.	Object
addedAt	The date and time of the description comment. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
text	The text of the description comment	String

##### [E] Examples

**Get problem description as an object:**



Client request:

```
GET /problems/9000000/description
X-API-Version: 1.0
Accept: application/json
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2011-12-05T14:49:21+0000",
    "addedBy": {
      "email": "nmellis@apple.com",
      "name": "Nathan Mellis"
    },
    "text": "There's chocolate in my peanut butter!"
  },
  {
    "addedAt": "2011-12-05T15:03:47+0000",
    "addedBy": {
      "email": "joel.young@apple.com",
      "name": "Joel Young"
    },
    "text": "There's peanut butter in my chocolate!"
  }
]
```

### Get problem description as an text:

Client request:

```
GET /problems/9000000/description
X-API-Version: 1.0
Accept: text/plain
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: text/plain; charset=utf-8

2011-12-05T14:49:21+0000 Nathan Mellis:
There's chocolate in my peanut butter!

2011-12-05T15:03:47+0000 Joel Young:
There's peanut butter in my chocolate!
```



### 4.1.2 Append To Problem Description

#### [A] Description

This API is used to post/append the description with timestamp and author. Only the text should be provided -- `addedBy` and `addedAt` fields are filled in according to the authenticated user and the creation time. On successful creation, the server responds with 201 `Created` status no response body.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
POST /problems/<problem_id>/description
```

#### [D] Attributes

Key	Description	Data Type
text	Contains description text to be added to a problem	String

#### [E] Examples

##### Append problem description:

Client request:

```
POST /problems/9000000/description
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "text": "There's chocolate in my peanut butter!"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

## 4.2 Problem Diagnosis

This group of APIs deals with fetching and appending to the problem diagnosis array.



### 4.2.1 Get Problem Diagnosis

#### [A] Description

This API is used to get/read the list of diagnosis previously added in a problem with timestamps and authors. If the user wants to fetch diagnosis as plain text, then 'Accept' key with value 'text/plain' needs to be passed as a header. If header is not passed then this API will fetch diagnosis as array.

The type parameter accepts the following values: "user", "history", "all". If "user" is specified, then only user entered text will be returned. If type is "history", the change history of the problem will be returned. "all" will return both types of data. If type is omitted, the default is "user".

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
GET /problems/<problem_id>/diagnosis
GET /problems/<problem_id>/diagnosis/<type>[,<type>]
```

#### [D] Response Attributes

Key	Description	Data Type
text	The text of the diagnosis comment.	String
addedAt	The date and time of the description comment. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
addedBy	The person who created the entry -- because of current data limitations, this will include only "name" and "email" attributes instead of the normal person attributes. In the future, this could be expanded.	Object

#### [D] Examples

##### Get problem diagnosis:

Client request:

```
GET /problems/9000000/diagnosis
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
```



```
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2011-12-05T14:49:21+0000",
    "addedBy": {
      "email": "nmellis@apple.com",
      "name": "Nathan Mellis"
    },
    "text": "There's chocolate in my peanut butter!"
  },
  {
    "addedAt": "2011-12-05T15:03:47+0000",
    "addedBy": {
      "email": "joel.young@apple.com",
      "name": "Joel Young"
    },
    "text": "There's peanut butter in my chocolate!"
  }
]
```

### Get problem diagnosis as an text:

**Note:-** If Accept Header is set to text/plain then date format will be change to MM/DD/YY HH:MM AM/PM GMT

Client request:

```
GET /problems/90000000/description
X-API-Version: 1.0
Accept: text/plain
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: text/plain; charset=utf-8

2011-12-05T14:49:21+0000 Nathan Mellis:
There's chocolate in my peanut butter!

2011-12-05T15:03:47+0000 Joel Young:
There's peanut butter in my chocolate!
```

## 4.2.2 Append To Problem Diagnosis

### [A] Description

This API is used to append diagnosis to a problem with timestamp and author. Only the text should be provided -- "addedBy" and "addedAt" fields are filled in according to the authenticated user and the



creation time. On successful creation, the server responds with 201 Created status and no response body.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

POST /problems/<problem\_id>/diagnosis

#### [D] Attributes

Key	Description	Data Type
text	Contains diagnosis text to be added to a problem	String

#### [E] Examples

##### Append problem diagnosis:

Client request:

```
POST /problems/9000000/diagnosis
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "text": "There's chocolate in my peanut butter!"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

## 4.3 Problem Enclosures

This group of APIs deals with the two enclosures collections, pictures and attachments. The APIs for the two are nearly identical, but they are accessed at two separate paths. Separate paths are necessary to differentiate between two files of the same name that are uploaded to different enclosure wells. Attachments also support full paths (which create subfolders) instead of flat filenames. Pictures do not support subfolders, but their ordering is significant.





### 4.3.1 Get Problem Enclosures List

#### [A] Description

This API provides a method to fetch enclosures using two different paths: one for attachments, and one for pictures. The enclosure contents can then be retrieved using [4.3.2 Download Problem Enclosure](#).

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
GET /problems/<problem_id>/attachments
GET /problems/<problem_id>/pictures
```

#### [D] Response Attributes

Key	Description	Data Type
fileName	The local path of the enclosure. Note: If the file uploaded contains "/" the slashes are converted to ":"	String
fileSize	The size of the enclosure in bytes.	Integer
addedAt	The date and time that the enclosure was added. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
addedBy	The user who uploaded the enclosure	Object

Below fields are added only in response of enclosures (Attachments not pictures).

Key	Description	Data Type
fileId	ID of Enclosure File.	String
createdAt	The date and time that the file was created. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
lastModifiedAt	The date and time that the file was last modified. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
privileges	Privileges of owner and user on file	Object



Key	Description	Data Type
encodeType	Encoding types used on enclosure file. It will contain all the encoding types used in comma separated string. This encoding is done if file is uploaded through Radar mac client. Web Services does not support encoding of any files at server. Various encode type used are BinHex, Encrypted, Sensitive, AppleSingle, Gzip	String or null

**Note about fileName:** if a file has been uploaded into a subdirectory, the fileName attribute will include the path to the file, relative to the collection name.

### [E] Examples

#### Get list of problem pictures:

Client request:

```
GET /problems/9000000/pictures
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2011-02-14T15:20:00+0000",
    "addedBy": {
      "dsid": 108039134,
      "email": "luke.burton@apple.com",
      "firstName": "Luke",
      "lastName": "Burton",
      "type": "Employee"
    },
    "fileName": "Pasted Picture 1",
    "fileSize": 7400
  },
  {
    "addedAt": "2011-02-14T15:35:00+0000",
    "addedBy": {
      "dsid": 102003482,
      "email": "jfarkas@apple.com",
      "firstName": "Jacob",
      "lastName": "Farkas",
      "type": "Employee"
    },
    "fileName": "failgail",
    "fileSize": 7400
  },
]
```



```
{
  "addedAt": "2011-02-14T16:57:00+0000",
  "addedBy": {
    "dsid": 12102825,
    "email": "mdimaggio@apple.com",
    "firstName": "Matt",
    "lastName": "DiMaggio",
    "type": "Employee"
  },
  "fileName": "Over9000",
  "fileSize": 7400
}
```

#### Get list of problem attachments:

Client request:

```
GET /problems/9000000/attachments
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2011-02-14T15:03:30+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileName": "Movies/IMG_0004.MOV",
    "fileSize": 107400,
    "fileId": "RJ1RDR0083761291",
    "createdAt": "2011-02-14T15:03:30+0000",
    "lastModifiedAt": "2011-02-14T15:03:30+0000",
    "encodeType": null,
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  }
]
```



```
    }
  }
]
```

### Get list of problem attachments (with subfolders):

Client request:

```
GET /problems/12311353/attachments
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2013-02-01T17:28:16+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447195",
    "fileName": "WebServicesTestFolder/Screen Shot 2013-02-01 at
8.13.22 AM.png",
    "fileSize": 38888,
    "createdAt": "2013-02-01T17:28:16+0000",
    "lastModifiedAt": "2013-02-01T17:28:16+0000",
    "encodeType": null,
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  },
  {
    "addedAt": "2013-02-01T17:28:16+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
```



```

      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447196",
    "fileName": "WebServicesTestFolder/Screen Shot 2013-02-01 at
8.13.20 AM.png",
    "fileSize": 44473,
    "createdAt": "2013-02-01T17:28:16+0000",
    "lastModifiedAt": "2013-02-01T17:28:16+0000",
    "encodeType": "Encrypted,Sensitive",
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  }
}
]

```

#### Get list of problem attachments (with subfolders that have slashes):

Client request:

```

GET /problems/12311353/attachments
X-API-Version: 1.0

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2013-02-01T17:32:18+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447201",
    "fileName": "Web:Services:Folder/Screen Shot 2013-02-01 at
8.13.22 AM.png",
    "fileSize": 38888,
    "createdAt": "2013-02-01T17:28:16+0000",
    "lastModifiedAt": "2013-02-01T17:28:16+0000",
    "encodeType": "BinHex,Encrypted,Sensitive",

```



```

    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  },
  {
    "addedAt": "2013-02-01T17:32:18+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447202",
    "fileName": "Web:Services:Folder/Screen Shot 2013-02-01 at
8.13.20 AM.png",
    "fileSize": 44473,
    "createdAt": "2013-02-01T17:32:18+0000",
    "lastModifiedAt": "2013-02-01T17:32:18+0000",
    "encodeType": "BinHex,Encrypted",
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  }
}
]

```

#### 4.3.2 Download Problem Enclosure

##### [A] Description

This API provides a method to download a specific attachment or picture, specified by path.

To download all attached file from enclosure in single call, a request need to be send with any filename with .zip extension and "X-Download-All: true" in the header.

Example /problems/12174826/attachments/test.zip



All the files will get downloaded as a zip file with the name passed in the request i.e test.zip

If request does not contain the file name with .zip extension ,then an error will be thrown "Please pass file name with .zip extension".

If 'X-Download-All' header is set to false or not pass in request than it will look for filename passed in request in enclosure and download specific file if found otherwise an error will be thrown saying file not attached to problem.

The filename passed in request should be URL encoded before sending, to download proper file from server.

## [B] Schedule

Required for version 1.0

## [C] URL Scheme

```
GET /problems/<problem_id>/attachments/<enclosure_path>
GET /problems/<problem_id>/pictures/<enclosure_path>
```

## [D] Examples

### Download A Specific Picture:

Client request:

```
GET /problems/9000000/pictures/Pasted%20Picture%201
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: image/jpeg
Content-Disposition: attachment; filename=Pasted Picture 1
<Binary Attachment Content>
```

### Download an Attachment from a Subfolder:

Client request:

```
GET /problems/9000000/attachments/Movies/IMG_0004.MOV
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
```



```
X-API-Version: 1.0
Status: 200
Content-Type: image/jpg
Content-Disposition: attachment; filename=IMG_0004.MOV
<Binary Attachment Content>
```

#### Download all Attachment from Enclosure:

Client request:

```
GET /problems/9000000/attachments/9000000.zip
X-Download-All: true
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: image/jpg
Content-Disposition: attachment; filename=9000000.zip
<Binary Attachment Content>
```

### 4.3.3 Upload Problem Enclosure

#### [A] Description

This API provides a means to upload an enclosure to either the attachments or pictures well. The enclosure path must be specified in the URL, with HTTP headers providing the content type. With attachments (but not pictures), the path can include directories, such as “subfolder/screenshot.png”, and any intermediate directories in the path will be created on the server.

Adding enclosures to a new problem is a special case: since multiple request bodies are not allowed, and since a user may not have access to a problem after its creation, certain access exceptions are given for enclosures on a newly created problem. Enclosures can be added to a problem, regardless of whether the user normally has write access to the problem, given the following conditions:

- The user requesting the enclosure upload is the originator
- The problem hasn’t been modified by anyone other than the originator

On success, the server responds with 201 Created and no response body.

To Override any files attached in attachments and pictures, `X-Override-File` header with value as `true` will need to be passed. It will override the existing attached file and upload the new file.

The filename passed in request should be URL encoded in-order to support special characters in filename.





## [B] Schedule

Required for version 1.0

## [C] URL Scheme

```
PUT /problems/<problem_id>/attachments/<enclosure_path>
PUT /problems/<problem_id>/pictures/<enclosure_path>
```

## [D] Examples

### Add a picture:

Client request:

```
PUT /problems/9000000/pictures/NewImage.jpg
X-API-Version: 1.0
Content-Type: image/jpg
<Binary Attachment Content>
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

### Add an attachment:

Client request:

```
PUT /problems/9000000/attachments/Movies/IMG_0004.MOV
X-API-Version: 1.0
Content-Type: video/quicktime
<Binary Attachment Content>
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

### Add an attachment (with X-Override-File):

Client request:

```
PUT /problems/9000000/attachments/Movies/IMG_0004.MOV
X-API-Version: 1.0
X-Override-File: true
Content-Type: video/quicktime
<Binary Attachment Content>
```

Server response:



```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### 4.3.4 Delete Problem Enclosure

##### [A] Description

This API provides a method to delete a specific enclosure, specified by path. On success, the server responds with 204 No Content. If the deleted enclosure is part of a subfolder, and the subfolder is empty after deletion, the folder will be deleted.

The filename passed in request should be URL encoded in-order to support special characters in filename.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
DELETE /problems/<problem_id>/attachments/<enclosure_path>
DELETE /problems/<problem_id>/pictures/<enclosure_path>
```

##### [D] Examples

###### Delete an enclosure:

Client request:

```
DELETE /problems/9000000/attachments/Subfolder/Attachment2.txt
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
X-API-Version: 1.0
Status: 204
```

#### 4.3.5 Modify Problem Enclosure

##### [A] Description

This API provides a method to modify the metadata for a specific image or document file -- currently, this just means renaming. The client passes the path to the enclosure, and the request body contains the new enclosure path. On success, the server returns a status of 200 Success.



The filename passed in request should be URL encoded in-order to support special characters in filename. The maximum length of filename is 255 characters.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
POST /problems/<problem_id>/attachments/<enclosure_path>
POST /problems/<problem_id>/pictures/<enclosure_path>
```

#### [D] Examples

##### Rename a Picture:

Client request:

```
POST /problems/9000000/pictures/OldImage.jpg
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "path": "NewImage.jpg"
}
```

Server response:

```
HTTP/1.1 200 Success
Status: 200
X-API-Version: 1.0
```

##### Rename an Attachment:

Client request:

```
POST /problems/9000000/attachments/Movies/IMG_0004.MOV
Content-Type: application/json; charset=UTF-8
X-API-Version: 1.0
{
  "path": "MyMovie.mov"
}
```

Server response:

```
HTTP/1.1 200 Success
Status: 200
X-API-Version: 1.0
```



### 4.3.6 Get Problem Picture Thumbnail

#### [A] Description

This API will be used to fetch the thumbnail of a picture by filename.

#### [B] Schedule

Required for version 1.5

#### [C] URL Scheme

```
GET /problems/<problem_id>/pictures/<filename>/thumbnail
```

#### [D] Examples

**Get list of problem pictures:**

Client request:

```
GET /problems/9000000/pictures/Pasted%20Picture%201/thumbnail
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: image/png

<Binary Image Content>
```

## 4.4 Related Problems

This set of APIs deals with relating other problems to a given Radar problem. A problem relation has a type, such as “related-to” or “blocking.” Since the same problem can be related multiple times in different ways, the URL path includes the relation type.

### 4.4.1 Get Related Problems List

#### [A] Description

This API provides a method to retrieve the list of related problems, including their types. The first URL will get all of the related problems for the given `problem_id`. The second URL will get all of the related problems for the given `problem_id` based on relation-type passed in the URL.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
GET /problems/<problem_id>/related-problems
```



```
GET /problems/<problem_id>/related-problems/<relation-type>[,<relation-type>]
```

#### [D] Response Attributes

Key	Description	Data Type
type	One of: "related-to", "original-of", "clone-of", "cloned-to", "blocked-by", "blocking", "parent-of", "subtask-of", "duplicate-of".	String
problem	Related problem object (see description in <a href="#">3.5 Problems Included in Other Objects</a> )	Object

#### [E] Examples

**Retrieve the list of related problems:**

Client request:

```
GET /problems/900000/related-problems
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "problem": {
      "assignee": {
        "dsid": 3795,
        "email": null,
        "firstName": "Brian",
        "lastName": "Lewis"
      },
      "component": {
        "name": "Alaska 98000",
        "version": "3.0"
      },
      "id": 2000000,
      "state": "Closed",
      "substate": "",
      "title": "Assigning the first bug!!!"
    },
    "relationType": "parent-of"
  },
  {

```



```

    "problem": {
      "assignee": {
        "dsid": 3795,
        "email": null,
        "firstName": "Brian",
        "lastName": "Lewis"
      },
      "component": {
        "name": "Radar 98000",
        "version": "3.0"
      },
      "id": 3000000,
      "state": "Closed",
      "substate": "",
      "title": "Assigning the Second bug!!!"
    },
    "relationType": "subtask-of"
  }
]

```

#### 4.4.2 Set Related Problems List

##### [A] Description

This API provides a method to APPEND to the list of related problems. On success, the server responds with 201 Created and no response body.

If an attempt is made to relate a problem/type that is already related, the request will succeed but no change will be made to the problem in Radar for that entry (i.e., no history will be added and the last modified date will not be updated).

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

POST /problems/<problem\_id>/related-problems

##### [D] Request Attributes

The request must be an array of objects with the following key:

Key	Description	Data Type
type	One of: "related-to", "blocked-by", "blocking", "parent-of", "subtask-of". Other relation types (such as "duplicate-of", "clone-of") are not allowed.	String



Key	Description	Data Type
problemID	Related problem ID	Integer

### [E] Examples

#### Append the list of related problems:

Client request:

```
POST /problems/900000/related-problems
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "problemID": 2000000,
    "type": "parent-of"
  },
  {
    "problemID": 3000000,
    "type": "subtask-of"
  }
]
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

### 4.4.3 Add Related Problem

#### [A] Description

This API provides a method to append a relation to the list of related problems. The request consists solely of a PUT and a URL path, without a request body. On success, the server responds with 201 Created and no response body.

If an attempt is made to relate a problem/type that is already related, the request will succeed but no change will be made to the problem in Radar for that entry (i.e., no history will be added and the last modified date will not be updated).

#### [B] Schedule

Required for version 1.0

**[C] URL Scheme**

```
PUT /problems/<problem_id>/related-problems/<relation_type>/<problem_id>
```

**[D] Request Attributes**

The request has no body. The `relation_type` in the url can be one of: "related-to", "blocked-by", "blocking", "parent-of", "subtask-of".

**[E] Examples****Relate a problem:**

Client request:

```
PUT /problems/900000/related-problems/parent-of/2000000
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

**4.4.4 Remove Related Problem****[A] Description**

This API provides a method to remove a relation from the list of related problems. The request consists solely of a DELETE and a URL path, without a request body. On success, the server responds with 204 No Content and no response body.

**[B] Schedule**

Required for version 1.0

**[C] URL Scheme**

```
DELETE /problems/<problem_id>/related-problems/<relation_type>/
<problem_id>
```

**[D] Request Attributes**

The request has no body. The `relation_type` in the url can be one of: "related-to", "blocked-by", "blocking", "parent-of", "subtask-of".

**[E] Examples****Remove a related problem:**





Client request:

```
DELETE /problems/900000/related-problems/parent-of/2000000
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```

#### 4.4.5 Edit Related Problem Relation

##### [A] Description

This API is used to change any existing related problem relation type.

The method will test the validity of the relation types and combinations of radar problem ID and relation types. The related problem must already exist with relation type mentioned in URL. A problem can not be related more than once with the same relation type. Relation Types must one of the supported values listed below. Appropriate error message will be returned for failed validations. Radar API must identify the relation to be changed by matching the combination of ID and relation type. A single ID can be related multiple times with different types.

Upon successful change of relation type for existing related problem ID, HTTP 200 OK will be returned in response without any response body.

Supported relation type enumerated values are "related-to", "blocked-by", "blocking", "subtask-of", "parent-of".

##### [B] Schedule

Required for version 1.5

##### [C] URL Scheme

```
PUT /problems/<problem_id>/related-problems/<related-problem-type>/
<related-problem-id>
```

##### [D] Request Parameters

Key	Description	Data Type	Required
relationType	Existing related Problem ID for which relation need to be changed.	Enumerated value	Y

**[E] Examples****Change existing related problem relation:**

Client request:

```
PUT /problems/14648918/related-problems/related-to/9053901
Content-Type: application/json; charset=UTF-8
{
  "relationType": "blocking"
}
```

Server response:

```
HTTP/1.1 200 Ok
Status: 200
```

**Error for incorrect existing relation type:**

Client request:

```
PUT /problems/14648918/related-problems/related-by/9053901
Content-Type: application/json; charset=UTF-8
{
  "relationType": "blocking"
}
```

Server response:

```
HTTP/1.1 404 Not Found
Status: 404
Content-Type: application/json; charset=utf-8
{
  "status": "404 Not Found",
  "title": "Relation Type Not Found",
  "message": "The problem ID of `9053901` does not have a `related-by`
relation type.",
  "help": "View documentation at http://radar.apple.com"
}
```

**Error for non-added related problem ID:**

Client request:

```
PUT /problems/14648918/related-problems/blocking/423423423
Content-Type: application/json; charset=UTF-8
{
  "relationType": "parent-of"
}
```

Server response:

```
HTTP/1.1 404 Not Found
```



```
Status: 404
Content-Type: application/json; charset=utf-8
{
  "status": "404 Not Found",
  "title": "Related Problem Not Found",
  "message": "The problem ID '423423423' is not related to '14648918'.",
  "help": "View documentation at http://radar.apple.com"
}
```

**Error for changing a relation type to an incorrect enumerated value:**

Client request:

```
PUT /problems/14648918/related-problems/related-to/9053901
Content-Type: application/json; charset=UTF-8
{
  "relationType": "blocking-of"
}
```

Server response:

```
HTTP/1.1 400 Bad Request
Status: 400
Content-Type: application/json; charset=utf-8
{
  "status": "400 Bad Request",
  "title": "Invalid Relation",
  "message": "The relation type 'blocking-of' is invalid.",
  "help": "View documentation at http://radar.apple.com"
}
```

**Error for changing a relation type to a non-supported enumerated value:**

Client request:

```
PUT /problems/14648918/related-problems/related-to/9053901
Content-Type: application/json; charset=UTF-8
{
  "relationType": "clone-of"
}
```

Server response:

```
HTTP/1.1 400 Bad Request
Status: 400
Content-Type: application/json; charset=utf-8
{
  "status": "400 Bad Request",
  "title": "Invalid Relation",
  "message": "The relation type 'clone-of' is invalid.",
  "help": "View documentation at http://radar.apple.com"
}
```



#### 4.4.6 Find Related Problems

##### [A] Description

This API provides a method to retrieve relationship tree for the problem. Request will contain two optional attributes and one mandatory attribute as mentioned in Request Attributes section.

Attribute 'type' will be an array of enumerated values. Types can contain all the values mentioned in 2.1.2 Relation Type enumeration except the basic related-to, for which relationship tree is not supported. Performing a find across multiple types will lead to performance degradation.

Attribute 'depth' will take integer value and describes depth of related problem tree need to be fetch in response. It is an optional value and defaults to 0.

Response will be an object containing different enumerated value passed in type attribute. Each enumerated value will internally contain its related problem tree object as an array. Please refer to example section for further reference.

##### [B] Schedule

Required for version 1.4.1

##### [C] URL Scheme

POST /problems/<problem\_id>/related-problems/find

##### [D] Request Attributes

Key	Description	Data Type	Required
type	Type of enumerated relationship type in an array. Enumeration are mentioned under 2.1.2 Relation Type enumeration.	Array of enumerated values	Y
depth	Depth of relationship tree	Integer	N

##### [E] Response Attributes

The response is an array of Related Problem Objects (as defined in "4.4 Related Problems" of the ERS) including additional fields on the problem to indicate if there are more related problems beyond the current "depth" search

Related Problem Object:



Key	Description	Data Type
type	An enumerated value indicating the relation type	Enumerated Value (String)
problem	The person who the problem is assigned to. See 7.2 People Included in Other Objects.	Related Problem Details Object

Related Problem Details Object:

Key	Description	Data Type
id	ID of the Problem	Integer
assignee	The person who the problem is assigned to. See 7.2 People Included in Other Objects.	Object
component	The problem's component. See 6.5 Components Included in Other Objects.	Object
title	Title of the Problem	String
state	State of the Problem	String
hasRelatedProblems	Flag to indicate whether or not related problems are present under the given tree object.	Boolean
relatedProblems	Relationship Tree of the Problem. Attributes are same as described in <a href="#">Response Attributes</a> table of Get Problem Relationship Tree section.	Array of Related Problem Object

## [E] Examples

Client request:

```
POST /problems/4179877/related-problems/find
Content-Type: application/json; charset=UTF-8
{
  "type":["subtask-of","parent-of"],
  "depth":1
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
{
  "subtask-of": [
    {
      "problem": {
```



```

    "assignee": {
      "dsid": 12345,
      "email": "jhackamack@apple.com",
      "firstName": "Jacob",
      "lastName": "Hackamack",
      "type": "Employee"
    },
    "component": {
      "name": "Alaska",
      "version": "Bloom Cty"
    },
    "hasRelatedProblems": true,
    "id": 14645688,
    "relatedProblems": [
      {
        "type": "subtask-of",
        "problem": {
          "assignee": {
            "dsid": 12345,
            "email": "jhackamack@apple.com",
            "firstName": "Jacob",
            "lastName": "Hackamack",
            "type": "Employee"
          },
          "component": {
            "name": "Alaska",
            "version": "Bloom Cty"
          },
          "hasRelatedProblems": false,
          "id": 14645628,
          "state": "Build",
          "title": "t%tv%.ilht"
        }
      }
    ],
    "state": "Analyze",
    "title": "t%tt%.ilht"
  },
  "type": "subtask-of"
}
],
"parent-of": [
  {
    "problem": {
      "assignee": {
        "dsid": 12345,
        "email": "jhackamack@apple.com",
        "firstName": "Jacob",
        "lastName": "Hackamack",
        "type": "Employee"
      },
      "component": {
        "name": "Alaska",
        "version": "Bloom Cty"
      }
    },

```



```

"hasRelatedProblems": true,
"id": 14645688,
"relatedProblems": [
  {
    "type": "parent-of",
    "problem": {
      "assignee": {
        "dsid": 12345,
        "email": "jhackamack@apple.com",
        "firstName": "Jacob",
        "lastName": "Hackamack",
        "type": "Employee"
      },
      "component": {
        "name": "Alaska",
        "version": "Bloom Cty"
      },
      "hasRelatedProblems": false,
      "id": 14645628,
      "state": "Build",
      "title": "t%tv%.ilht"
    }
  },
  {
    "state": "Analyze",
    "title": "t%tt%.ilht"
  },
  {
    "type": "parent-of"
  }
]
}

```

## 4.5 Problem Keywords

This set of APIs deals with relating keywords to a given Radar problem. In addition to the keyword itself, a keyword relation has an author and creation time. Since a keyword with the same name can exist in different components, the user has the option to specify keywords by either id (which is guaranteed to work with a valid id) or by name (which may cause a conflict even with a valid name).

### 4.5.1 Get Problem Keywords List

#### [A] Description

This API provides a method to retrieve the list of keywords, including their authors and creation times.

#### [B] Schedule

Required for version 1.0



## [C] URL Scheme

GET /problems/<problem\_id>/keywords

## [D] Response Attributes

Key	Description	Data Type
keyword	Related keyword object (see description in <a href="#">5.3 Keywords Included in Other Objects</a> )	Object
addedBy	The person who added the keyword relation (see description in <a href="#">7.2 People Included in Other Objects</a> )	Object
addedAt	The date and time that the keyword was added.	ISO 8601 date string

## [E] Examples

### Retrieve the list of keywords:

Client request:

```
GET /problems/900000/keywords
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2011-02-16T19:55:00+0000",
    "addedBy": {
      "dsid": 108039134,
      "email": "luke.burton@apple.com",
      "firstName": "Luke",
      "lastName": "Burton",
      "type": "Employee"
    },
    "keyword": {
      "id": 85959,
      "name": "Angry Birds Hitlist"
    }
  },
  {
    "addedAt": "2011-02-16T06:15:00+0000",
    "addedBy": {
```





```
        "dsid": 78395640,
        "email": "jgale@apple.com",
        "firstName": "John",
        "lastName": "Gale",
        "type": "Employee"
    },
    "keyword": {
        "id": 81852,
        "name": "I Want A Pony"
    }
},
{
    "addedAt": "2011-02-14T23:20:00+0000",
    "addedBy": {
        "dsid": 108039134,
        "email": "luke.burton@apple.com",
        "firstName": "Luke",
        "lastName": "Burton",
        "type": "Employee"
    },
    "keyword": {
        "id": 52846,
        "name": "Snakes on a Radar"
    }
},
{
    "addedAt": "2011-12-16T01:09:00+0000",
    "addedBy": {
        "dsid": 78395640,
        "email": "jgale@apple.com",
        "firstName": "John",
        "lastName": "Gale",
        "type": "Employee"
    },
    "keyword": {
        "id": 96580,
        "name": "This is not a Test"
    }
},
{
    "addedAt": "2011-02-15T03:06:00+0000",
    "addedBy": {
        "dsid": null,
        "email": null,
        "firstName": "Tom",
        "lastName": "Duffy",
        "type": null
    },
    "keyword": {
        "id": 83305,
        "name": "why we can't have nice things"
    }
}
}
```

1



## 4.5.2 Set Problem Keywords List

### [A] Description

This API provides a method to APPEND the list of keywords for a problem. If some of the added keywords are already related to the problem, they will not be changed (i.e. the addedAt and addedBy values will remain the same). On success, the server responds with 201 Created and no response body.

If a keyword is specified as a string, the searches for a matching keyword among Personal, Shared (subscribed), and public keywords (defined within the component). If nothing is found, the search then expands to all keywords visible in radar. Closed keywords are ignored. If no visible keywords are found in all of Radar, the server responds with 404 Not Found. If multiple keywords match, the server responds with 409 Conflict.

### [B] Schedule

Required for version 1.0

### [C] URL Scheme

```
POST /problems/<problem_id>/keywords
```

### [D] Request Attributes

The request must be an array of keyword strings or ids.

### [E] Examples

#### Append the list of keywords:

Client request:

```
POST /problems/900000/keywords
X-API-Version: 1.0
[
  "why we can't have nice things",
  "I Want a Pony"
]
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```



### 4.5.3 Add Keyword to Problem

#### [A] Description

This API provides a method to append a keyword to the list of keywords related to a problem. The request consists solely of a `PUT` and a URL path, without a request body. On success, the server responds with `201 Created` and no response body.

A note on parsing the keyword value: if it is solely numeric, it is considered to be an id, not a name. So if there is a keyword called “123” and a keyword with id 123, the request will add the keyword with the id of 123 rather than the keyword “123”.

If the client specifies the keyword by name and it isn’t unique, it should respond with `409 Conflict`.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
PUT /problems/<problem_id>/keywords/<keyword_name>
PUT /problems/<problem_id>/keywords/<keyword_id>
```

#### [D] Examples

##### Add a keyword by name to a problem:

Client request:

```
PUT /problems/900000/keywords/I%20Want%20a%20Pony
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

##### Add a keyword by name, with conflict:

Client request:

```
PUT /problems/900000/keywords/Test
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 409 Conflict
Status: 409
X-API-Version: 1.0
```



```
Content-Type: application/json;charset=UTF-8
{
  "help": "Please email radar-help@group.apple.com",
  "message": "The keyword you are attempting to add is not unique by name. Please use the keyword id",
  "status": "409 Conflict",
  "title": "Keyword name not unique"
}
```

#### 4.5.4 Remove Problem Keyword

##### [A] Description

This API provides a method to remove a keyword from a problem. The request consists solely of a DELETE and a URL path, without a request body. On success, the server responds with 204 No Content and no response body.

A note on parsing the keyword value: if it is solely numeric, it is considered to be an id, not a name. So if there is a keyword called "123" and a keyword with id 123, the request will remove the keyword with the id of 123 rather than the keyword "123".

If the client specifies the keyword by name and there are multiple keywords with that name attached to the problem, it will remove all such keywords.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
DELETE /problems/<problem_id>/keywords/<keyword_name>
DELETE /problems/<problem_id>/keywords/<keyword_id>
```

##### [E] Examples

###### Delete a keyword attached to a problem:

Client request:

```
DELETE /problems/900000/keywords/123
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
X-API-Version: 1.0
Status: 204
```



#### 4.5.5 Get Problem Keyword

##### [A] Description

This API is used to look up a particular keyword association. If the keyword is associated with the problem, its information will be returned, and if not, the server responds with a 404 Not Found status.

A note on parsing the keyword value: if it is solely numeric, it is considered to be an id, not a name. So if there is a keyword called "123" and a keyword with id 123, the request will return the keyword with the id of 123 rather than the keyword "123".

If multiple keywords with the given name are attached to the problem, the server responds with 409 Conflict.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
GET /problems/<problem_id>/keywords/<keyword_name>
GET /problems/<problem_id>/keywords/<keyword_id>
```

##### [D] Response Attributes

The response object has the same attributes as 4.5.1 Get Problem Keywords List.

##### [E] Examples

**Check whether a keyword attached to a problem:**

Client request:

```
GET /problems/9000000/keywords/Snakes%20on%20a%20Radar
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "addedAt": "2011-02-14T23:20:00+0000",
  "addedBy": {
    "dsid": 108039134,
    "email": "luke.burton@apple.com",
    "firstName": "Luke",
    "lastName": "Burton",
    "type": "Employee"
  }
}
```



```

    },
    "keyword": {
        "id": 52846,
        "name": "Snakes on a Radar"
    }
}

```

#### 4.5.6 Add multiple Keywords to Active/Inactive Problem

##### [A] Description

This API is used to add multiple keywords to multiple inactive or active problems. This API will be newly created as WebService 1.0 APIs accepts KeywordNames and single problemID . The keyword in request must contains the ID of the keyword.

Adding keyword to inActive problem does not bring problem in active state and modification history will not be updated in diagnosis and email notifications will not be triggered.

On a successful response, the HTTP status is 201 Created. If keyword ID or problem ID not found, the HTTP status is 400 Bad Request.

The error message format will be as follows where 'xxxxx' will be Problem ID. The same error message will be shown for invalid keywordID.

"Problem ID xxxxxx does not exists"

##### [B] Schedule

Required for version 1.1

##### [C] URL Scheme

POST /problems/keywords/add

##### [D] Request Attributes

Request will be array of objects containing below attributes. Request array can have maximum of 100 objects otherwise it will result in connection timeout. The request object will be same for addition and removing of keywords.

Key	Description	Data Type
problemID	Problem ID to which keywords has to be attached	Integer
keywordIDs	Array of keyword IDs which has to be attached to problem. Array can have maximum of 5 keyword IDs.	Array of Integers

**[E] Examples****Adding multiple keywords to multiple problems:**

Client request:

```
POST /problems/keywords/add
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  {
    "keywordIDs": [
      130013,
      130014,
      67941,
      67939,
      70053
    ],
    "problemID": 300000
  },
  {
    "keywordIDs": [
      49599,
      46601
    ],
    "problemID": 400000
  }
]
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

**4.5.7 Remove multiple Keywords from Active/Inactive Problem****[A] Description**

This API is used to remove multiple keywords from multiple active/inactive problems. This API will be newly created as WebService 1.0 APIs accepts KeywordNames and single problemID . The keyword in request must contains the ID of the keyword.

Removing keyword from inActive problem does not bring problem in active state and modification history will not be updated in diagnosis and email notifications will not be triggered.

On a successful response, the HTTP status is 204. If keyword ID or problem ID not found, the HTTP status is 400.



The error message format will be as follows where 'xxxxx' will be Problem ID. The same error message will be shown for invalid keywordID.

"Problem ID xxxxxx does not exists"

#### [B] Schedule

Required for version 1.1

#### [C] URL Scheme

PUT /problems/keywords/remove

#### [D] Request Attributes

Request will be array of objects containing below attributes. Request array can have maximum of 100 objects otherwise it will result in connection timeout. The request object will be same for addition and removing of keywords.

Key	Description	Data Type
problemID	Problem ID to which keywords has to be attached	Integer
keywordIDs	Array of keyword IDs which has to be attached to problem. Array can have maximum of 5 keyword IDs.	Array of Integers

#### [E] Examples

##### Removing multiple keywords from multiple problems:

Client request:

```
PUT /problems/keywords/remove
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
  {
    "keywordIDs": [
      130013,
      130014,
      67941,
      67939,
      70053
    ],
    "problemID": 300000
  },
  {
    "keywordIDs": [
      49599,
```





```

        46601
      ],
      "problemID": 400000
    }
  ]

```

Server response:

```

HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0

```

## 4.6 Other Related Items

This set of APIs deals with relating “other items” to a given Radar problem.

### 4.6.1 Get Other Related Items List

#### [A] Description

This API provides a method to retrieve the list of related items.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
GET /problems/<problem_id>/other-related-items
```

#### [D] Response Attributes

Key	Description	Data Type
id	ID of the related item	String or null
system	Name of the system that the related item comes from. (See list of system values in <a href="#">4.6.2 Get Other Related Items Systems List</a> )	String or null
title	Local title of the related item	String or null
url	URL for external system	String

#### [E] Examples

**Retrieve the list of other related items:**

Client request:



```
GET /problems/9000000/other-related-items
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "id": "41601",
    "system": "Espresso",
    "title": "Request to get the webServer certs",
    "url": "EXP2://Ticket/41601"
  },
  {
    "id": "52918",
    "system": "Sonar",
    "title": "Encoding issue with emails having foreign characters",
    "url": "sonr://request/52918"
  }
]
```

**Retrieve the list of other related items for multiple problems:**

Client request:

```
GET /problems/12345723,12345722/other-related-items
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "id": 12345723,
    "otherRelatedItems": [
      {
        "id": "11424",
        "title": "test",
        "system": "Knowledge Base",
        "url": "http://support.apple.com/kb/11424"
      }
    ]
  },
  {
    "id": 12345722,
    "otherRelatedItems": [
      {
        "id": "312312",

```



```

        "title": "Testing",
        "system": "Dialog",
        "url": "dlog://log/312312"
    }
  ]
}
]

```

#### 4.6.2 Get Other Related Items Systems List

##### [A] Description

This API provides a method to retrieve the list of possible values for “system”, along with whether it’s a global system or a user-specific one.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

GET /other-related-items/systems

##### [D] Response Attributes

Key	Description	Data Type
name	The name of the system	String
isGlobal	Is the system global, or defined in the user’s preferences?	Boolean
urlScheme	The url scheme for items in this system	String

##### [E] Examples

###### Get Other Related Items Systems list:

Client request:

```

GET /other-related-items/systems
X-API-Version: 1.0

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8

```



```
X-API-Version: 1.0
[
  {
    "isGlobal": true,
    "name": "Radar Scheduled Test",
    "urlScheme": "rdar://st/<id>"
  },
  ...
]
```

#### 4.6.3 Set List of Other Related Items

##### [A] Description

This API provides a method to append to the list of other related items. The existing relations will be retained. On success, the server responds with 201 Created and no response body.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
POST /problems/<problem_id>/other-related-items
```

##### [D] Request Attributes

The request is an array of objects with the following keys:

Key	Description	Data Type
id	ID of the other related item	String
system	Name of the system	String
title	Local title of the related item (optional)	String

##### [E] Examples

###### Append to the list of other related items:

Client request:

```
POST /problems/9000000/other-related-items
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
```



```
    "id": "43721",
    "system": "Sonar",
    "title": "Ability to set assignee during Convert to
Correspondence"
  },
  {
    "id": "52712",
    "system": "Espresso",
    "title": "Ability to set assignee during Convert to
Correspondence"
  }
]
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### 4.6.4 Add Other Related Item

##### [A] Description

This API provides a method to add an other related item. The request consists of a PUT and a URL path, and an optional title for the item. On success, the server responds with 201 Created and no response body.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
PUT /problems/<problem_id>/other-related-items/<system_name>/<item_id>
```

##### [D] Request Attributes

Since the `system_name` and `item_id` are supplied in the url, the only request attribute remaining is the title. If no title is supplied, the request body can be skipped altogether.

Key	Description	Data Type
title	Local title of the related item	String

##### [E] Examples

Add an external relation:



Client request:

```
PUT /problems/900000/other-related-items/Sonar/43721
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "title": "...
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### 4.6.5 Edit Other Related Item

##### [A] Description

This API provides a method to edit an other related item. The request consists of a POST and a URL path, and an mandatory title for the item. On success, the server responds with 201 Created and no response body.

##### [B] Schedule

Required for version 1.4

##### [C] URL Scheme

```
POST /problems/<problem_id>/other-related-items/<system_name>/<item_id>
```

##### [D] Request Attributes

Key	Description	Data Type
title	Title of the related item	String

##### [E] Examples

**Edit an external relation:**

Client request:

```
POST /problems/900000/other-related-items/Sonar/43721
Content-Type: application/json;charset=UTF-8
{
  "title": "Changing Title"
}
```



Server response:

```
HTTP/1.1 201 Created
Status: 201
```

#### 4.6.6 Remove Other Related Item

##### [A] Description

This API provides a method to remove an other related item. The request consists solely of a **DELETE** and a URL path, without a request body. On success, the server responds with **204 No Content** and no response body.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
DELETE /problems/<problem_id>/other-related-items/<system_name>/
<item_id>
```

##### [D] Request Attributes

The request URL consists of the system name and the item id of the related item that is to be removed.

##### [E] Examples

###### Remove an external relation:

Client request:

```
DELETE /problems/9000000/other-related-items/Sonar/43721
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```

## 4.7 Third Party Products

This set of APIs deals with adding, removing and fetching the third party product information for a Radar problem.



#### 4.7.1 Get Third Party Product List

##### [A] Description

This API provides a method to retrieve the list of third party products attached to the problem.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

GET /problems/<problem\_id>/third-party-products

##### [D] Response Attributes

Key	Description	Data Type
name	Name of the third party product attached	String
version	Version of the third party product attached	String
publisher	Name of the publisher for the third party product	String

##### [D] Examples

**Retrieve the list of third party products:**

Client request:

```
GET /problems/900000/third-party-products
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "name": "Radiant",
    "publisher": "Hexage,Ltd",
    "version": "3.5"
  },
  {
    "name": "Solitaire",
    "publisher": "Byterun,Ltd",

```





```
        "version": "1.2"  
    }  
]
```

#### 4.7.2 Set Third Party Product List

##### [A] Description

This API provides a method to APPEND to the list of third party products. On success, the server responds with 201 Created and no response body.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
POST /problems/<problem_id>/third-party-products
```

##### [D] Request Attributes

The request must be an array of objects with the following keys:

Key	Description	Data Type
name	Name of the third party product	String
version	Version of the third party product	String
publisher	Name of the publisher for the third party product	String

##### [E] Examples

###### Append the list of third party products:

Client request:

```
POST /problems/900000/third-party-products  
X-API-Version: 1.0  
Content-Type: application/json; charset=utf-8  
[  
  {  
    "name": "RapidWeaver",  
    "publisher": "Hexage,Ltd",  
    "version": "5.0"  
  },  
  {  
    "name": "Radium",
```



```
        "publisher": "CatPig Studios Inc.",  
        "version": "2.7"  
    }  
]
```

Server response:

```
HTTP/1.1 201 Created  
Status: 201  
X-API-Version: 1.0
```

### 4.7.3 Add Third Party Product

#### [A] Description

This API provides a method to append a product to the list of third party products added for the problems. The request consists solely of a PUT and a URL path, without a request body. On success, the server responds with 201 Created and no response body.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
PUT /problems/<problem_id>/third-party-products/<publisher>/<name>/  
    <version>
```

#### [D] Examples

##### Add a third party product:

Client request:

```
PUT /problems/900000/third-party-products/Universalis%20Publishing  
    %20Ltd./Catholic%20Calendar/1.4  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 201 Created  
Status: 201  
X-API-Version: 1.0
```



#### 4.7.4 Remove Third Party Product

##### [A] Description

This API provides a method to remove a third party product from the list of third party products added to the problems. The request consists solely of a DELETE and a URL path, without a request body. On success, the server responds with 204 No Content and no response body.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
DELETE /problems/<problem_id>/third-party-products/<publisher>/<name>/<version>
```

##### [D] Examples

###### Remove a third party product from a problem:

Client request:

```
DELETE /problems/900000/third-party-products/Universalis%20Publishing%20Ltd./Catholic%20Calendar/1.4
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```

#### 4.7.5 Find Third Party Product

##### [A] Description

This API provides a method to search for Third Party product. The response of this API contains the array of object satisfying request parameters. Request should contain any one of attribute mentioned under Request parameters section as a search criteria otherwise an error message will be shown.

##### [B] Schedule

Required for version 1.5

##### [C] URL Scheme

```
POST /third-party-products/find
```



### [D] Request Parameters

The Request body should contain any of the below parameters.

Parameter	Description	Data Type
name	Third Party Product name	String
version	Third party product version	String
publisher	Third Party Product publisher	String
platform	World name This seems to be the platform name from the example - if so, then it should be named 'platform'	String

### [D] Response Parameters

Response will be array of object containing below attributes.

Parameter	Description	Data Type
name	Third Party Product name	String
version	Third party product version	String
publisher	Third Party Product publisher	String
platform	World name This seems to be the platform name from the example - if so, then it should be named 'platform'	String

### [E] Examples

**Get all Third Party Product with name AOL:**

Client request:

```
GET /third-party-products/find
Content-Type: application/json;charset=UTF-8
{
  "name": "AOL"
}
```

Server response:

```
HTTP/1.1 200 Ok
Status: 200
Content-Type: application/json; charset=UTF-8
[
```



```

{
  "name": "AOL",
  "platform": "Macintosh",
  "publisher": "AOL",
  "version": "1.0"
},
{
  "name": "AOL-2",
  "platform": "Macintosh",
  "publisher": "AOL",
  "version": "2.0"
}
.... more objects
]

```

## 4.8 Security List

This set of APIs deals with adding, removing and fetching the security list for a Radar problem. Each item on the list is a person, work group, or access group, and this person/group is granted Assignable privileges on the problem (regardless of the individual access levels of the people within a work group).

For the purposes of these APIs, the Security List is just the editable part of the list shown in the Radar GUI. The read-only component groups shown in the GUI are not considered part of the Security List here.

### 4.8.1 Get Security List

#### [A] Description

This API provides a method to retrieve the security list for a problem.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
GET /problems/<problem_id>/security-list
```

#### [D] Response Attributes

Key	Description	Data Type
type	Type of person or group: one of "Person", "Work Group", or "Access Group".	String



Key	Description	Data Type
status	Status of person or group. (See <a href="#">10.1 Get Field Enumeration</a> for possible values.)	String
isInherited	'isInherited' to security list response will contain true for security person/group inherited from component and false for others	boolean
object	The person, work group, or access group. For a person, see <a href="#">7.2 People Included in Other Objects</a> . For groups, this will consist of "id" and "name" key.	Object

## [D] Examples

### Retrieve the security list:

Client request:

```
GET /problems/13196386/security-list
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "status": "External",
    "isInherited": true,
    "object": {
      "id": 18918,
      "name": "DS: CLIwata Directs Emp 56 5818$"
    },
    "type": "Access Group"
  },
  {
    "status": "External",
    "isInherited": true,
    "object": {
      "id": 20682,
      "name": "Exilant Developers"
    },
    "type": "Access Group"
  }
]
```



## 4.8.2 Add Person or Group to Security List

### [A] Description

This API provides a method to append a person or group to the security list for a problem. The request consists solely of a PUT and a URL path, without a request body. On success, the server responds with 201 Created and no response body.

### [B] Schedule

Required for version 1.0

### [C] URL Scheme

```
PUT /problems/<problem_id>/security-list/<type>/<id_or_name>
```

### [D] Request Attributes

The Request URL contains the problem\_id, the type of the person or group, and the dsid (for a person) or name (for a group).

Type can be "person", "work-group" or "access-group".

### [E] Examples

#### Add a person to the security list:

Client request:

```
PUT /problems/900000/security-list/person/8794
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### Add an access group to the security list:

Client request:

```
PUT /problems/900000/security-list/access-group/RAM%20DAC%20All
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```



### 4.8.3 Remove Person or Group from Security List

#### [A] Description

This API provides a method to remove a person or group from the security list for a problem. The request consists solely of a DELETE and a URL path, without a request body. On success, the server responds with 204 No Content and no response body.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
DELETE /problems/<problem_id>/security-list/<type>/<id_or_name>
```

#### [D] Request Attributes

The Request URL contains the problem\_id, the type of the person or group, and the dsid (for a person) or name (for a group).

Type can be “people”, “work-groups” or “access-groups”.

#### [E] Examples

##### Remove a person from the security list:

Client request:

```
DELETE /problems/900000/security-list/people/8794
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```

##### Remove an access group from the security list:

Client request:

```
DELETE /problems/900000/security-list/access-groups/RAM%20DAC%20A11
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```





#### 4.8.4 Append to Security List

##### [A] Description

This API provides a method to append a list of people or groups to the security list for a problem. On success, the server responds with 201 Created and no response body.

##### [B] URL Scheme

```
POST /problems/<problem_id>/security-list
```

##### [C] Request Attributes

The request consists of an array of objects with the following attributes:

Key	Description	Data Type
type	The type of the person or group to be added. Type can be "person", "work-group" or "access-group".	String
dsid	If type is "person", this will be the DSID of the person. If type is not "person", this will be ignored.	Integer
name	If type is "work-group" or "access-group", this will be the name of the group. If type is "person", this will be ignored.	String

##### [D] Response Attributes

The response will be the entire security list, including any entries that have been added, as described in [4.8.1 Get Security List](#).

##### [E] Examples

###### Append to the security list:

Client request:

```
POST /problems/9000000/security-list
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  {
    "dsid": 8794,
    "type": "person"
  },
  {
    "name": "RAM DAC All",
    "type": "access-group"
  }
]
```



```
}
]
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### 4.8.5 Get Problem Privileges

##### [A] Description

This API provides a method to get the privilege of a user for a given problem or problems.

##### [B] Schedule

Required for version 1.3

##### [C] URL Scheme

```
GET /security/<dsid>/problems/<problem_id>[,<problem_id>]
```

##### [D] Response Attributes

The response consists of an array of objects with the following attributes:

Key	Description	Data Type
problemID	ID of the problem	Integer
personDSID	DSID of the person	Integer
problemPrivilege	Privilege of user for the problem	String

##### [E] Examples

##### Get Problem Privileges :

Client request:

```
GET /security/1118581234/problems/12310128,12310737
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
```



```
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "problemID": 12310128,
    "personDSID": 1118581234,
    "problemPrivilege": "Super User"
  },
  {
    "problemID": 12310737,
    "personDSID": 1118581234,
    "problemPrivilege": "Super User"
  }
]
```

## 4.9 CC List

This set of APIs deals with the list of people who are CCed on a problem.

### 4.9.1 Get CC List

#### [A] Description

This API provides a method to retrieve the list of people who are CCed on a problem, including their visibility into the problem.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
GET /problems/<problem_id>/cc-list
```

#### [D] Response Attributes

Key	Description	Data Type
person	The person who is CCed. See <a href="#">7.2 People Included in Other Objects</a> .	Object
isProblemVisible	Indicates whether this person is able to view the problem under its current access permissions. If not, the person will not receive CC notifications.	Boolean

**[E] Examples****Retrieve the CC List:**

Client request:

```
GET /problems/900000/cc-list
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "isProblemVisible": true,
    "person": {
      "dsid": 123,
      "email": "joel.young@apple.com",
      "firstName": "Joel",
      "lastName": "Young",
      "type": "Employee"
    }
  },
  ...
]
```

**4.9.2 Add Person to CC List****[A] Description**

This API provides a method to add a person to the CC list. The person must be specified by DSID, which can be found using [7.1 Find People](#). On success, the server responds with 201 Created and no response body.

**[B] Schedule**

Required for version 1.0

**[C] URL Scheme**

```
PUT /problems/<problem_id>/cc-list/<dsid>
```

**[D] Request Attributes**

The URL contains the DSID which can be found using [7.1 Find People](#).



## [E] Examples

### Add a person to the CC List:

Client request:

```
PUT /problems/900000/cc-list/123
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

## 4.9.3 Remove Person from CC List

### [A] Description

This API provides a method to remove a person from the CC list. The request consists solely of a DELETE and a URL path, without a request body. On success, the server responds with 204 No Content and no response body.

### [B] Schedule

Required for version 1.0

### [C] URL Scheme

```
DELETE /problems/<problem_id>/cc-list/<dsid>
```

### [D] Examples

#### Remove the related problem:

Client request:

```
DELETE /problems/900000/cc-list/123
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```

## 4.10 Target Milestones

This set of APIs deals with the list of target milestones for a problem.



#### 4.10.1 Get Target Milestones List

##### [A] Description

This API provides a method to retrieve the list of milestone targets for a problem.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
GET /problems/<problem_id>/target-milestones
```

##### [D] Response Attributes

If the target is an existing problem, it will contain all of the following attributes; otherwise, if it is a placeholder, it will have only milestone (which includes component).

Key	Description	Data Type
milestone	The targeted milestone.	Object
problem	The target problem.	Problem Object

##### [E] Examples

###### Retrieve the Target Milestones List:

Client request:

```
GET /problems/900000/target-milestones
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "milestone": {
      "component": {
        "name": "Purple",
        "version": "1.0"
      },
      "name": "Durango"
    }
  },
  ...
]
```



```

{
  "milestone": {
    "component": {
      "name": "Mac OS",
      "version": "X"
    },
    "name": "Barolo"
  },
  "problem": {
    "assignee": {
      "dsid": 12345,
      "email": "ewalt@apple.com",
      "firstName": "Alan",
      "lastName": "Ewalt",
      "type": "Employee"
    },
    "component": {
      "name": "Radar",
      "version": "New Bugs"
    },
    "id": 7654321,
    "state": "Analyze"
  }
}
]

```

#### 4.10.2 Add Target Milestone to Problem

##### [A] Description

This API provides a method to add a target milestone to a problem. The target can be specified as either a problem or a milestone (which will generate a placeholder). On success, the server responds with 201 Created and no response body.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

POST /problems/<problem\_id>/target-milestones

##### [D] Request Attributes

Key	Description	Data Type
milestone	The targeted milestone. The placeholder's component will be inferred from the milestone. (See <a href="#">6.3.3 Component Milestones Included in Other Objects</a> )	Object



Key	Description	Data Type
problem	The problem targeted for the milestone. The only required attribute is ID.	Object

### [E] Examples

#### Add a problem to target milestones list:

Client request:

```
POST /problems/900000/target-milestones
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "problem": {
    "id": 1000000
  }
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### Add a placeholder to target milestones list:

Client request:

```
POST /problems/900000/target-milestones
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "milestone": {
    "component": {
      "name": "Mac OS",
      "version": "Zin"
    },
    "name": "Zin"
  }
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```





### 4.10.3 Remove Target Milestone from Problem

#### [A] Description

This API provides a method to remove a problem or a placeholder from the target milestones list. The target can be removed as either a problem or a milestone (which will remove a placeholder) On success, the server responds with 204 No Content and no response body.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
DELETE /problems/<problem_id>/target-milestones
```

#### [D] Request Attributes

Key	Description	Data Type
milestone	The targeted milestone. The placeholder's component will be inferred from the milestone. (See <a href="#">6.3.3 Component Milestones Included in Other Objects</a> )	Object
problem	The problem targeted for the milestone. The only required attribute is ID.	Object

#### [E] Examples

**Remove a placeholder from the target milestones list:**

Client request:

```
DELETE /problems/900000/target-milestones
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "milestone": {
    "component": {
      "name": "Mac OS",
      "version": "Zin"
    },
    "name": "Zin"
  }
}
```

Server response:



```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```

#### Remove a problem from the target milestones list:

Client request:

```
DELETE /problems/900000/target-milestones
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "problem": {
    "id": 100000
  }
}
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```

#### 4.10.4 Clone Problem to Target Milestones

##### [A] Description

This API provides a method to convert a placeholder target into a problem target (using the clone). The server takes a request body and creates a single clone according to the given milestone. On success, the server returns the problem that was created.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
POST /problems/<problem_id>/target-milestones/clone
```

##### [D] Request Attributes

The milestone must be specified.

Key	Description	Data Type
milestone	The targeted milestone. The placeholder's component will be inferred from the milestone. (See <a href="#">6.3.3 Component Milestones Included in Other Objects</a> )	Object



Key	Description	Data Type
reason	Reason text for cloning milestone. It is a optional attribute. Maximum length is 1100 characters.	String

### [E] Examples

#### Clone a problem to a placeholder:

Client request:

```
POST /problems/900000/target-milestones/clone
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "milestone": {
    "component": {
      "name": "Mac OS",
      "version": "Zin"
    },
    "name": "Zin"
  }
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
Content-Type: application/json; charset=utf-8
X-API-Version: 1.0
{
  "component": {
    "name": "Mac OS",
    "version": "Zin"
  },
  "id": 12345678,
  "title": "...",
  ...
}
```

#### Clone a problem to a placeholder with reason text:

Client request:

```
POST /problems/900000/target-milestones/clone
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "milestone": {
    "component": {
      "name": "Mac OS",
```



```

        "version": "Zin"
      },
      "name": "Zin"
    }
    "reason": "Cloned Target Milestone Zin for tracking"
  }

```

Server response:

```

HTTP/1.1 201 Created
Status: 201
Content-Type: application/json; charset=utf-8
X-API-Version: 1.0
{
  "component": {
    "name": "Mac OS",
    "version": "Zin"
  },
  "id": 12345678,
  "title": "...",
  ...
}

```

#### 4.10.5 Edit Target Milestones Placeholders

##### [A] Description

This API is used to edit the target milestone placeholder.

The old milestone and component name and version should be passed in URL The request body should contain the new milestone name with the component name and version to which it belongs.

Validation should be done to check whether the placeholder passed in URL is already attached to problem or not. If not then an error message will be returned.

On Success HTTP 204 status will be return without any response body.

##### [B] Schedule

Required for version 1.2

##### [C] URL Scheme

```

PUT    /problems/<problem-id>/target-milestones/components/<component-
name>/<component-version>/milestones/<milestone>

```

**[D] Request Attributes**

Key	Description	Data Type
component	A Component Object	Component Object
milestone	Name of the new milestone	String

**[E] Examples**

Client request:

```
PUT    /problems/8630711/target-milestones/components/Radar/Automation/
milestones/RadarWS-1.1
{
  "component":{"name": "Radar",
  "version": "Dev-Automation"},
  "milestone": "TBD",
}
```

Server response:

```
HTTP/1.1 204 Created
Status: 204
```

## 4.11 Problem History

This set of APIs deals with retrieving a problem's history, which is an audit trail of a subset of problem attributes.

### 4.11.1 Get Problem History

**[A] Description**

This API provides a method to retrieve a problem's history, which is an audit trail of a subset of problem attributes. Each entry in the history is a set of changes to the assignee, state, component, milestone, and/or priority: the changes are listed together for atomicity when searching over the history.

**[C] URL Scheme**

```
GET /problems/<problem_id>/history
```

**[D] Response Attributes**



Key	Description	Data Type
changedBy	The person who made the change. (See <a href="#">7.2 People Included in Other Objects</a> )	Object
changedAt	The date and time of the modification.	Timestamp
assignee	The person the problem was assigned to after the change. (See <a href="#">7.2 People Included in Other Objects</a> )	Object
previousState	The problem's state before the change. See <a href="#">10.1 Get Field Enumeration</a> for a list of possible values.	Enumerated String
changedState	The problem's state after the change. See <a href="#">10.1 Get Field Enumeration</a> for a list of possible values.	Enumerated String
proxyUser	The person who is proxy of the problem	Object
assignorsResponseCode	Previous assignee response code	Integer
whenReadDate	Problem read date by assignee	Timestamp
workTimeQuantity	Work time Quantity of the problem	Integer
isReadByAssignee	Boolean flag to indicate problem read by assignee	Boolean
responseCode	Response code for the problem	Integer
whenReassignedDate	TimeStamp when problem reassigned to assignee	Timestamp
assignedType	Problem assigned type	String or null
assignmentType	Problem Assignment type	String or null
component	The problem's component after the change. (See <a href="#">6.7 Components Included in Other Objects</a> )	Object
changedMilestone	The problem's component milestone after the change. (See <a href="#">6.3.3 Component Milestones Included in Other Objects</a> )	Object
previousMilestone	The problem's component milestone before the change. (See <a href="#">6.3.3 Component Milestones Included in Other Objects</a> )	Object
changedPriority	The problem's priority after the change.	Integer
previousPriority	The problem's priority before the change.	Integer
changedEvent	The problem's component event after change.	String or null
previousEvent	The problem's component event before change.	String or null



Key	Description	Data Type
changedComponentID	ComponentID after the change.	Integer
previousComponentID	ComponentID after the change.	Integer

## [E] Examples

### Retrieve the Problem History:

Client request:

```
GET /problems/900000/history
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "assignee": {
      "dsid": 327472,
      "email": "jem@apple.com",
      "firstName": "Jerry",
      "lastName": "Majors Patterson",
      "type": "Employee"
    },
    "changedAt": "2011-02-14T13:50:00Z",
    "changedBy": {
      "dsid": 187179976,
      "email": "kartik@apple.com",
      "firstName": "Kartik",
      "lastName": "Vaithyanathan",
      "type": "Employee"
    },
    "changedState": "Analyze",
    "component": {
      "name": "CPU Software Yellow",
      "version": "New Bugs"
    },
    "milestone": null,
    "previousState": "Verify",
    "priority": 5
  },
  ...
]
```



#### 4.11.2 Get Problem Diagnosis History

##### [A] Description

This API provides a method to retrieve history of a problem or multiple problems by specifying the problem ID or IDs in the url.

For example , if a relation is added or removed, then response will be displayed as

```
{
  "addedRelation": "This problem is a clone of radar: / / 1234123",
  "removedRelation": "This problem is no longer parent of radar: / / 123445",
}
```

On a successful response, the HTTP status is 200. When an error occurs, an appropriate HTTP response code is used, along with a JSON representation of the error.

##### [B] Schedule

Required for version 1.2

##### [C] URL Scheme

GET problems/history/<problem\_id>[,<problem\_id>]

##### [D] Response Attributes

The response attributes will contain the below attributes.

Key	Description	Data Type
id	ID of the problem for which history is fetched.	Integer
history	history will contain the array of below objects.	Array

#### Attributes of history object

Key	Description	Data Type
author	Name of the person who modified the problem.	String
addedAt	The date and time of the history change.	ISO 8601 datetime string
action	action will contain the below objects.	Object

#### Attributes of action Object





Key	Description	Data Type
assignee	Assignee name before and after change (Refer Table1.2 for format)	Object
title	Title of problem before and after change (Refer Table1.2 for format)	Object
component	Component of problem before and after change (Refer Table1.2 for format)	Object
resolution	Resolution of the problem before and after change (Refer Table1.2 for format)	Object
classification	Classification of problem before and after change (Refer Table1.2 for format)	Object
priority	Priority of problem before and after change (Refer Table1.2 for format)	Object
proxy	Proxy of problem before and after change (Refer Table1.2 for format)	Object
fixOrdering	Fix ordering of problem before and after change (Refer Table1.2 for format)	Object
state	State of problem before and after change (Refer Table1.2 for format)	Object
substate	Substate of problem before and after change (Refer Table1.2 for format)	Object
reproducible	Reproducibility of problem before and after change (Refer Table1.2 for format)	Object
currentDateNeeded	Current date needed for problem before and after change (Refer Table1.2 for format)	Object
originalDateNeeded	Original date needed for problem before and after change (Refer Table1.2 for format)	Object
targetStartDate	Start Date of target before and after change (Refer Table1.2 for format)	Object
currentTargetCompletionDate	Current completion date of target before and after change (Refer Table1.2 for format)	Object



Key	Description	Data Type
originalTargetCompletionDate	original completion date of target before and after change (Refer Table1.2 for format)	Object
currentTotalEstimateOfEffort	Current total estimate of effort before and after change (Refer Table1.2 for format)	Object
originalTotalEstimateOfEffort	Original total estimate of effort before and after change (Refer Table1.2 for format)	Object
percentageOfEffortCompleted	Percentage of total effort completed before and after change (Refer Table1.2 for format)	Object
remainingEffort	Remaining effort before and after change (Refer Table1.2 for format)	Object
expendedEffort	Expended effort before and after change (Refer Table1.2 for format)	Object
deliveryVehicle	Delivery Vehicle before and after change (Refer Table1.2 for format)	Object
testCaseID	ID of test case before and after change (Refer Table1.2 for format)	Object
taskOrder	Order of task before and after change (Refer Table1.2 for format)	Object
targetMilestonesMasterID	Target milestone masterID before and after change (Refer Table1.2 for format)	Object
targetMilestone	Target milestone before and after change (Refer Table1.2 for format)	Object
externalSystemTitle	External system title before and after change (Refer Table1.2 for format)	Object
configSummary	Config summary before and after change (Refer Table1.2 for format)	Object
serialNumber	Serial number before and after change (Refer Table1.2 for format)	Object
foundInBuild	Found in build before and after change (Refer Table1.2 for format)	Object



Key	Description	Data Type
fixedinBuild	Fixed in build before and after change (Refer Table1.2 for format)	Object
verifiedInBuild	Verified in build before and after change (Refer Table1.2 for format)	Object
mustBeFixedByBuild	Must be fixed by build before and after change (Refer Table1.2 for format)	Object
originator	Originator of problem before and after change (Refer Table1.2 for format)	Object
dri	DRI of problem before and after change (Refer Table1.2 for format)	Object
resolver	Resolver of problem before and after change (Refer Table1.2 for format)	Object
epm	EPM of problem before and after change (Refer Table1.2 for format)	Object
securityType	Security type of problem before and after change (Refer Table1.2 for format)	Object
cwePrimary	CWE(Primary) of product security before and after change (Refer Table1.2 for format)	Object
cweSecondary	CWE(Secondary) of product security before and after change (Refer Table1.2 for format)	Object
isPrivacy	Is Privacy of product security before and after change (Refer Table1.2 for format)	Object
securityColor	Security color of product security before and after change (Refer Table1.2 for format)	Object
attackVector	Attack Vector of product security before and after change (Refer Table1.2 for format)	Object
assets	Assests of product security before and after change (Refer Table1.2 for format)	Object
complexity	Complexity of product security before and after change (Refer Table1.2 for format)	Object



Key	Description	Data Type
exploitability	Exploitability of product security before and after change (Refer Table1.2 for format)	Object
impact	Impact of product security before and after change (Refer Table1.2 for format)	Object
userBase	User Base of product security before and after change (Refer Table1.2 for format)	Object
visibility	Visibility of product security before and after change (Refer Table1.2 for format)	Object
securityDri	Security DRI of product security before and after change (Refer Table1.2 for format)	Object
authentication	Authentication of product security before and after change (Refer Table1.2 for format)	Object
securityVerifier	Security Verifier of product security before and after change (Refer Table1.2 for format)	Object
reporterName	Reporter Name of product security before and after change (Refer Table1.2 for format)	Object
reporterEmail	Reporter Email of product security before and after change (Refer Table1.2 for format)	Object
sonar	Sonar of Reporter before and after change (Refer Table1.2 for format)	Object
seed	Seed of Reporter before and after change (Refer Table1.2 for format)	Object
credit	Credit of Reporter before and after change (Refer Table1.2 for format)	Object
coordinate	Coordinate of Reporter before and after change (Refer Table1.2 for format)	Object
externalId	External Id of product security before and after change (Refer Table1.2 for format)	Object



Key	Description	Data Type
problemDescription	Changes made to problem description field.	String
keywordAdded	Name of the keyword added to the problem.	String
keywordRemoved	Name of the keyword removed from the problem.	String
pictureAdded	Name of the picture added to the problem	String
enclosure	Changes made to Enclosure	String
enclosureCount	Changes made to enclosure count	String
addedRelation	Relation added to the problem	String
relationTypeChange	Relation changed for the problem	String
removedRelation	Relation removed for the problem	String
targetMilestonesAdded	Target Milestone added for the problem	String
targetMilestonesRemoved	Target Milestone removed for the problem	String
targetMilestonesPlaceholderAdded	Target Milestone Placeholder added for the problem	String
targetMilestonesPlaceholderRemoved	Target Milestone Placeholder removed for the problem	String
externalSystemsAdded	External system added for the problem	String
externalSystemsRemoved	External system removed for the problem	String
configuration	Changes made to configuration	String
workaround	Changes made to workaround	String
releaseNotes	Changes made to releaseNotes	String
sourceChanges	Changes made to sources Change	String
rootCause	Changes made to root changes	String
correctiveAction	Changes made to corrective Action	String



Key	Description	Data Type
thirdPartyProductAdded	Name of the third party product added	String
thirdPartyProductRemoved	Name of the third party product removed	String
failedModuleAdded	Name of the failed module added	String
failedModuleRemoved	Name of the failed module removed	String
detailOfFailureAdded	Detail of failure added	String
detailOfFailureRemoved	Detail of failure removed	String
targetAdded	Target added to product security	String
targetRemoved	Target removed for the product security	String
externalIdAdded	External ID added for the product security	String
externalIdRemoved	External ID removed for the product security	String
problemSecurityAdded	Problem Security added for the problem	String
problemSecurityRemoved	Problem Security removed for the problem	String
readByAssignee	Read By assignee check box	String
approved	Approved check box	String
umbrella	umbrella check box	String
isAutoCalculated	Auto-Calculate Dates and Efforts from subtask check box	String
newAPI	New API check box	String
3rdPartyContent	3rd party content check box	String
needsPatentReview	Needs Patent Review check box	String
newSPI	New SPI check box	String
importOrExport	Import/Export check box	String



Key	Description	Data Type
confidentialContent	Confidential content check box	String
hl	Hl check box	String
loc	LOC check box	String
hasOpenSource	Has Open Source check box	String
verifiedByTester	Verified By Tester check box	String
mustBeRegressed	Must Be regressed check box	String
criticalToFix	Critical To Fix check box	String
data	If the history text does not matches any of the pattern, then it will be displayed as it with "data" as a key.	String Array

**Table 1.2. Attributes for changed Field.**

Key	Description	Data Type
from	This will contain the previous value of the changed field.	String
to	This will contain the changed value of the field.	String

### [E] Examples

Client request:

```
GET /problems/history/900000,100000
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "id" : 900000,
    "history" : [
      {
        "author" : "radar tester",
        "addedAt" : "2011-12-21T11:39:00-0800",
        "action" :
        {
```



```

    "addedRelation": "This problem is a clone of radar://1234123",
    "removedRelation" : "This problem is no longer parent of radar://
123445",
    "thirdPartyProductAdded": "radium",
  }
},
{
  "author" : "radar tester",
  "addedAt" : "2011-12-21T11:40:00-0800",
  "action" :
  {
    "problemDescription": "New information added",
    "priority": {
      "from": "5 - Not Set",
      "to": "1 - Showstopper"
    },
    "assignee": {
      "from": "User X",
      "to": "User Y",
    },
    "targetMilestone": {
      "from": "",
      "to": "Milestone X"
    }
  }
},
{
  "id" : 100000,
  "history" : [
    {
      "author" : "radar tester",
      "addedAt" : "2011-12-21T11:39:00-0800",
      "action" :
      {
        "keywordAdded": "key",
        "targetMilestonesAdded" : "Problem id: rdar://problem/10061807
with milestone",
      }
    },
    {
      "author" : "radar tester",
      "addedAt" : "2011-12-21T11:40:00-0800",
      "action" :
      {
        "pictureAdded": "picture",
        "readByAssignee" : "checked",
        "approved " : "checked",
        "state": {
          "from": "Analyze",
          "to": "Verify"
        },
        "title": {
          "from": "Test",

```





```

        "to": "Test1",
      },
      "securityColor": {
        "from": "Red",
        "to": "Green"
      }
    }
  }
]
}
]

```

## 4.12 Problem Crash APIs

This set of APIs deals with crash attribute of the problem.

### 4.12.1 Update Problems Crash Count

#### [A] Description

This API is used to update the Radar Crash Count for multiple problems. It can update the crash count of inactive problem without bringing them back to active. On successful update, HTTP status is 201 Created.

Adding crashCount to inActive problem does not bring problem in active state and modification history will not be updated in diagnosis and email notifications will not be triggered.

The error message format will be as follows where 'xxxxx' will be Problem ID.  
 "Problem ID xxxxxx does not exists"

#### [B] Schedule

Required for version 1.1

#### [C] URL Scheme

POST /problems/crashCount

#### [D] Request Attributes

Request will contain an array of below two attributes. The array can have maximum of 100 records otherwise it will result in connection timeout.

Key	Description	Data Type
problemID	Problem ID for which crash count need to update.	Integer
crashCount	Crash count which need to update to problem	Integer

**[E] Examples**

Client request:

```
POST /problems/crashCount
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
  {
    "crashCount": 100,
    "problemID": 300000
  },
  {
    "crashCount": 150,
    "problemID": 400000
  },
  {
    "crashCount": 200,
    "problemID": 500000
  }
]
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

**4.12.2 Get Crash Tracer Radar List****[A] Description**

This API is used to get the list of Radar's or Problem's having keyword "crashtracer-originated" or "crashtracer-tracking".

By Default the rowlimit is set to 2000. To increase the row limit a header 'X-rowlimit' need to be set in request. Using curl command header can be specified as shown below.

-H "X-rowlimit:row\_limit\_value", where row\_limit\_value is a number.

As the result set of this API is very large, If the result not came within 10 mins then server timeout will happen showing 502 HTTP code.

The request URL contain single attribute "state". It should contain either active or inactive to get active or inactive problem ids in response. state attribute refers to problem active and inactive state.

**[B] Schedule**

Required for version 1.1

**[C] URL Scheme**

```
GET /problems/crash-tracer-radars/<state>
```

**[D] Response Attributes**

The Response will be in JSON Object format and contains below two attributes. The “result ” attribute will contain a JSON Array object containing Crash Radar ProblemID mentioned below.

Key	Description	Data Type
ids	Array containing Crash Tracer Radar problemID	Array of Integers
resultRows	Number of Problems fetched	Integer

**[E] Examples****Get Crash Tracer Radar List:**

Client request:

```
GET /problems/crash-tracer-radars/active
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: application/json;charset=UTF-8
{
  "ids": [
    6860092,
    10163652,
    5677577
  ],
  "resultRows": 3
}
```

**4.12.3 Get Crash Duplicate Chain****[A] Description**

This API is used to fetch the crash tracer Duplicate chain. The request contains an array of problemIDs which has been returned from API Get Crash Tracer Radar List.



The API result set is very large. If the results take longer than 10 minutes to respond the server will display the HTTP code 502 Bad Gateway.

#### [B] Schedule

Required for version 1.1

#### [C] URL Scheme

POST /problems/crash-dup-chain

#### [D] Request Attributes

Key	Description	Data Type
problemIDs	Array of problemIDs	Array of Integers

#### [D] Response Attributes

The Response will be in JSON Object format and contains below two attributes. The “result ” attribute will contain a JSON Array object containing Crash Radar Object mentioned below.

Key	Description	Data Type
result	Array containing Crash Radar object	Array of objects
resultRows	Number of rows fetched	Integer

#### Crash Radar Object

Key	Description	Data Type
affectedProblemId	Original problem ID	Integer
affectsProblemId	Affected problem ID	Integer or null
crashCount	Crash count for the problem	Integer or null

#### [E] Examples

##### Get Crash Tracer Radar List:

Client request:

POST /problems/crash-dup-chain



```
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "problemIDs": [
    6860092,
    5716309,
    10163652
  ]
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "result": [
    {
      "affectedProblemId": 6860092,
      "affectsProblemId": null,
      "crashCount": null
    },
    {
      "affectedProblemId": 6860092,
      "affectsProblemId": 5716309,
      "crashCount": 210
    },
    {
      "affectedProblemId": 10163652,
      "affectsProblemId": 10154678,
      "crashCount": null
    },
    {
      "affectedProblemId": 5677577,
      "affectsProblemId": null,
      "crashCount": 35
    },
    ...
  ],
  "resultRows": 10
}
```

## 4.13 Product Security Targets

This section has APIs to get, add, remove and edit the targets of Product Security for a problem.

### 4.13.1 Get Targets for Problem

#### [A] Description

This API is used to get all the Product Security target attached for a particular problem.

**[B] Schedule**

Required for version 1.2

**[C] URL Scheme**

GET /problems/<problem\_id>/product-security/targets

**[D] Response Attributes**

The Response will be array of objects containing below attributes. If no targets has been attached for a problem then an empty array ( [ ] )will be returned.

Key	Description	Data Type
id	ID of the target	Integer
affectedProduct	Affected product Object	Object or null
recommendedReleaseVehicle	Recommended release vehicle Object	Object or null
plannedReleaseVehicle	Planned release vehicle Object	Object or null
trackingProblemID	Id of the related tracking problem	Integer or null
isCriticalToFix	Is product critical to fix	Boolean
isVerified	Is target verified by PS	Boolean or null
component	A component Object	Component Object
trackingProblemMilestone	A milestone object for tracking problem	Milestone Object or null
isDescribed	Is Target Describe?	Boolean

**[E] Examples**

Client request:

GET /problems/9000000/product-security/targets

Server response:

HTTP/1.1 200 OK

Status: 200

Content-Type: application/json; charset=utf-8

```
[{
  "id":472,
  "affectedProduct":{"id":151,
    "name": "Radar-7.0"},
  "recommendedReleaseVehicle":{"id":null,
    "name":null},
```



```

    "plannedReleaseVehicle":{"id":null,
    "name":null},
    "trackingProblemID":null,
    "isCriticalToFix":false,
    "isVerified":null,
    "component":{
        "name":null,
        "version":null
    },
    "trackingProblemMilestone":null,
    "isDescribed": true
},
{
    "id":487,
    "affectedProduct":{"id":155,
    "name": "Radar-7.1"},
    "recommendedReleaseVehicle":{"id":210,
    "name": "Radar Automation"},
    "plannedReleaseVehicle":{"id":211,
    "name": "Radar WebService"},
    "trackingProblemID":null,
    "isCriticalToFix":true,
    "isVerified":true,
    "component":{"name": "Radar",
    "version": "Automation"},
    "trackingProblemMilestone":{"name": "RadarWS",
    "component":{
        "name": "Radar",
        "version": "Automation"
    }},
    "isDescribed": false
}
}]

```

#### 4.13.2 Add Target to Problem

##### [A] Description

This API is used to add Product Security target for a particular problem. The target request must contain either 'affectedProduct' or 'trackingProblem' attribute and all other attributes are non-mandatory for adding a target to problem.

An Error message will be returned if the 'trackingProblem' or combination of "affectedProduct" and "recommendedReleaseVehicle" target is already attached to problem.

On Success 201 Created will be returned and an error message will be returned if any of the passed attribute is invalid.

##### [B] Schedule

Required for version 1.2

##### [C] URL Scheme



POST /problems/<problem\_id>/product-security/targets

#### [D] Request Attributes

The Request must contain either 'affectedProduct' or 'trackingProblem' for adding a target to problem.

Key	Description	Data Type
affectedProduct	Name of the affected product	String
recommendedReleaseVehicle	Name of the recommended release vehicle	String
plannedReleaseVehicle	Name of the planned release vehicle	String
trackingProblem	Id of the related tracking problem	Integer
isCriticalToFix	Is product critical to fix. Default false will be set if not passed in request.	Boolean
isVerified	Is target verified by PS. Default 'null' will be set if not passed in request.	Boolean or null
isDescribed	Is Target Describe?	Boolean

#### [E] Examples

Client request:

```
POST /problems/9000000/product-security/targets
{
  "affectedProduct": "AirPort Base Station",
  "isCriticalToFix": true,
  "isVerified": null,
  "isDescribed": true
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
```

### 4.13.3 Edit Target of Problem

#### [A] Description

This API is used to edit Product Security target for a particular problem.

The id should be passed in URL to identify the target which has to be modified and id can be fetch by using 3.13.3.1 Get Targets for Problem. If id is invalid or not found then an error message will be returned.





On Success 201 Created will be returned and an error message will be returned if any of the passed attribute is invalid.

#### [B] Schedule

Required for version 1.2

#### [C] URL Scheme

```
PUT /problems/<problem_id>/product-security/targets/<target-id>
```

#### [D] Request Attributes

**Request Attribute is same as mentioned in section 3.13.3.2 Add Target to Problem Request Attribute.**

#### [E] Examples

Client request:

```
PUT /problems/9000000/product-security/targets/472
{
  "product": 157,
  "criticalToFix": true,
  "verifiedByPS": -1,
  "isDescribed": true
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
```

### 4.13.4 Remove Target of Problem

#### [A] Description

This API is used to remove Product Security target for a particular problem.

The rowID should be passed in URL to identify the target which has to be remove and rowID can be fetch by using 3.13.3.1 Get Targets for Problem. If rowID is invalid or not found then an error message will be returned.

On Success 204 No Content will be returned and an error message will be returned if any of the passed attribute is invalid or not found.

#### [B] Schedule

Required for version 1.2

#### [C] URL Scheme

```
DELETE /problems/<problem_id>/product-security/targets/<rowID>
```

**[D] Examples**

Client request:

`DELETE /problems/9000000/product-security/targets/472`

Server response:

`HTTP/1.1 204 No Content``Status: 204`**4.14 Product Security Reporters**

This section has APIs to get, add, remove and edit the Reporters of Product Security for a problem.

**4.14.1 Get Reporters for Problem****[A] Description**

This API is used to get all the Product Security Reporter attached for a particular problem.

If user does not have privileges to see the reporters for a problem then an error message will be returned.

**[B] Schedule**

Required for version 1.2

**[C] URL Scheme**`GET /problems/<problem_id>/product-security/reporters`**[D] Response Attributes**

The Response will be array of objects containing below attributes. If no reporters has been attached for a problem then an empty array ( `[]` )will be returned.

Key	Description	Data Type
id	Row ID of the reporter	Integer
name	Name of the Reporter	String or null
email	Email of the Reporter	String or null
sonar	Sonar ID related to Reporter	String or null
isSeed	Seed of the reporter	Boolean or null
isCredit	Credit of the Reporter	Boolean or null
isCoordinate	Coordinate of the Reporter	Boolean or null

**[E] Examples**

Client request:

```
GET /problems/9000000/product-security/reporters
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "isCoordinate": false,
    "isCredit": true,
    "email": "john@smith.com",
    "id": 123,
    "name": "John Smith",
    "isSeed": false,
    "sonar": "1234"
  },
  {
    "isCoordinate": false,
    "isCredit": true,
    "email": "john@smith.com",
    "id": 123,
    "name": "John Smith",
    "isSeed": null,
    "sonar": "1234"
  }
]
```

**4.14.2 Add Reporter to Problem****[A] Description**

This API is used to add Product Security reporter for a particular problem. The reporter request must contain either 'name' or 'email' attribute and all other attributes are non-mandatory for adding a reporter to problem.

An Error message will be returned if combination of "name" and "email" attribute reporter is already attached to problem.

On Success 201 Created will be returned and an error message will be returned if any of the passed attribute is invalid.

**[B] Schedule**

Required for version 1.2

**[C] URL Scheme**



POST /problems/<problem\_id>/product-security/reporters

#### [D] Request Attributes

The Request must contain either 'name' or 'email' for adding a reporter to problem.

Key	Description	Data Type
name	Name of the Reporter	String
email	Email of the Reporter	String
sonar	Sonar ID related to Reporter	String
isSeed	Seed code of the Reporter.	Boolean or null
isCredit	Credit code of the Reporter.	Boolean or null
isCoordinate	Coordinate code of the Reporter.	Boolean or null

#### [E] Examples

Client request:

```
POST /problems/9000000/product-security/reporters
{
  "name": "Remote Reporter",
  "seed": true,
  "credit": false
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
```

### 4.14.3 Edit Reporter of Problem

#### [A] Description

This API is used to edit Product Security reporter for a particular problem.

The rowID should be passed in URL to identify the reporter which has to be modified and reporter-ID can be fetch by using 3.13.4.1 Get Reporters for Problem. If reporter-ID is invalid or not found then an error message will be returned.

On Success 201 Created will be returned and an error message will be returned if any of the passed attribute is invalid.

#### [B] Schedule



Required for version 1.2

#### [C] URL Scheme

```
PUT /problems/<problem_id>/product-security/reporters/<reporter-ID>
```

#### [D] Request Attributes

All the below fields are non-mandatory. The field passed in request will be updated and others are unchanged. If request does not contain any attribute than an error message will be returned.

Request Attribute is same as mentioned in section 3.13.4.2 Add Reporters to Problem Request Attribute.

#### [E] Examples

Client request:

```
PUT /problems/9000000/product-security/reporters/238
{
  "isCoordinate": null,
  "isCredit": true,
  "email": "john@smith.com",
  "name": "John Smith",
  "isSeed": false,
  "sonar": "1234"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
```

### 4.14.4 Remove Reporter of Problem

#### [A] Description

This API is used to remove Product Security reporter for a particular problem.

The rowID should be passed in URL to identify the reporter which has to be removed and reporter-ID can be fetch by using 3.13.4.1 Get Reporters for Problem. If reporter-ID is invalid or not found then an error message will be returned.

On Success 204 No Content will be returned and an error message will be returned if any of the passed attribute is invalid or not found.

#### [B] Schedule

Required for version 1.2

#### [C] URL Scheme

```
DELETE /problems/<problem_id>/product-security/reporters/<reporter-ID>
```

#### [D] Examples



Client request:

```
DELETE /problems/9000000/product-security/reporters/238
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
```

## 4.15 Product Security ExternalIDs

This section has APIs to get, add, remove and edit the external information for a problem.

### 4.15.1 Get ExternalIDs for Problem

#### [A] Description

This API is used to get all the Product Security externals attached for a particular problem.

If user does not have privileges to see the externalID for a problem then an error message will be returned.

#### [B] Schedule

Required for version 1.2

#### [C] URL Scheme

```
GET /problems/<problem_id>/product-security/externals
```

#### [D] Response Attributes

The Response will be array of objects containing below attributes. If no externalIDs has been attached for a problem then an empty array ( [ ] )will be shown.

Key	Description	Data Type
id	External ID	Integer
external	The name of the external resource	String

#### [E] Examples

Client request:

```
GET /problems/9000000/product-security/externals
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
```



```

        "external": "Radar",
        "id": 215
    },
    {
        "external": "129846",
        "id": 216
    }
]

```

#### 4.15.2 Add ExternalID to Problem

##### [A] Description

This API is used to add Product Security externals for a particular problem.

An Error message will be returned if the external reference attribute is already attached to problem.

On Success 201 Created will be returned.

##### [B] Schedule

Required for version 1.2

##### [C] URL Scheme

```
POST /problems/<problem_id>/product-security/externals
```

##### [D] Request Attributes

Key	Description	Data Type
external	The name of the external relation	String

##### [E] Examples

Client request:

```

POST /problems/9000000/product-security/externals
{
  "external": "Reporter"
}

```

Server response:

```

HTTP/1.1 201 Created
Status: 201

```

#### 4.15.3 Edit ExternalID of Problem

##### [A] Description



This API is used to edit Product Security externals for a particular problem.

The rowID should be passed in URL to identify the externals which has to be updated and external-ID can be fetch by using 3.13.5.1 Get ExternalIDs for Problem. If external-ID is invalid or not found then an error message will be returned.

On Success 201 Created will be returned and an error message will be returned if external-ID passed in URL is not found.

#### **[B] Schedule**

Required for version 1.2

#### **[C] URL Scheme**

```
PUT /problems/<problem_id>/product-security/externals/<externals-id>
```

#### **[D] Request Attributes**

Request Attribute is same as mentioned in section 3.13.5.2 Add ExternalID to Problem Request Attribute.

#### **[E] Examples**

Client request:

```
PUT /problems/9000000/product-security/externals/1032
{
  "external": "10341978"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
```

### **4.15.4 Remove ExternalID of Problem**

#### **[A] Description**

This API is used to remove Product Security externals for a particular problem.

The rowID should be passed in URL to identify the externals which has to be remove and external-ID can be fetch by using 3.13.5.1 Get ExternalIDs for Problem. If external-ID is invalid or not found then an error message will be returned.

On Success 204 No Content will be returned and an error message will be returned if external-ID not found.

#### **[B] Schedule**

Required for version 1.2



**[C] URL Scheme**

```
DELETE /problems/<problem_id>/product-security/externals/<externals-id>
```

**[D] Examples**

Client request:

```
DELETE /problems/9000000/product-security/externals/1032
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
```

**4.16 Product Security Product****4.16.1 Get Products List****[A] Description**

This API is used to get all the Product Security Management Products. If user do not have privileges to manage Product Security then an error message will be returned.

**[B] Schedule**

Required for version 1.2

**[C] URL Scheme**

```
GET /product-security/products
```

**[D] Response Attributes**

The Response will be array of objects containing below attributes. If no product has been defined then an empty array ( [ ] )will be returned.

Key	Description	Data Type
id	ID of the product	Integer
name	Name of the product	String
isClosed	Is product closed.?	Boolean

**[E] Examples**

Client request:

```
GET /product-security/products
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
```



```
Content-Type: application/json; charset=utf-8
[{
  "id": 231,
  "name": "iWork (iOS)",
  "isClosed": false
},
{
  "id": 232,
  "name": "iTunes Stores",
  "isClosed": false
},
... more records
]
```

#### 4.16.2 Create Product

##### [A] Description

This API is used to create a new Product Security Product. If user does not have privileges to manage Product Security then an error message will be returned.

##### [B] Schedule

Required for version 1.3

##### [C] URL Scheme

POST /product-security/products

##### [D] Request Attributes

Key	Description	Data Type
name	Name of the product	String
isClosed	Is product closed?	Boolean
isRestricted	Is the product restricted?	Boolean
groupNames	Access or Work groups to restrict access to members of the added groups	Array of Strings

##### [E] Response Attributes

Created product object will be returned with response.



Key	Description	Data Type
id	ID of the product	Integer
name	Name of the product	String
isClosed	Is product closed?	Boolean
isRestricted	Is the product restricted?	Boolean
groups	Access or Work groups to restrict access to members of the added groups. See <a href="#">Product Security Group</a> table for description.	Array of group object

### Product Security Group

Key	Description	Data Type
id	ID of the group	Integer
name	Name of the group	String
type	Type of the group. Either Access or Work.	String

### [F] Example

Client request:

```
POST /product-security/products
{
  "name": "iWork (Mac)",
  "isClosed": false,
  "isRestricted": true,
  "groupNames": [
    "radar-automation"
  ]
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.3
{
  "id": 72,
  "isRestricted": true,
  "name": "iWork (Mac)",
  "isClosed": false,
  "groups": [
    {
      "id": 19790,
      "name": "radar-automation",
      "type": "Access"
    }
  ]
}
```



```
    }
  ]
}
```

### 4.16.3 Open Product

#### [A] Description

This API is used to open a specified Product Security Product. If user does not have privileges to manage Product Security then an error message will be returned.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

```
GET /product-security/products/<product-id>
```

#### [D] Response Attributes

Response attributes are same as described in [Response Attributes](#) of section 4.16.2 Create a Product Security Product.

#### [E] Example

Client request:

```
GET /product-security/products/72
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.2
{
  "id": 72,
  "isRestricted": true,
  "name": "iPhone",
  "isClosed": false,
  "groups": [
    {
      "id": 19790,
      "name": "Phone Orpheus",
      "type": "Access"
    },
    {
      "id": 637,
      "name": "Test Casers",
      "type": "Work"
    }
  ]
}
```



#### 4.16.4 Update Product

##### [A] Description

This API is used to update an existing Product Security Product. If user does not have privileges to manage Product Security then an error message will be returned.

##### [B] Schedule

Required for version 1.3

##### [C] URL Scheme

```
PUT /product-security/products/<product-id>
```

##### [D] Request Attributes

Request attributes are same as described in [Request Attributes](#) of section 4.16.2 Create a Product Security Product.

##### [E] Example

Client request:

```
PUT /product-security/products/72
{
  "name": "iWork (Mac)",
  "isClosed": false,
  "isRestricted": true,
  "groupNames": [
    "radar-automation"
  ]
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.3
```

#### 4.16.5 Delete Product

##### [A] Description

This API is used to delete a Product Security Product. If user does not have privileges to manage Product Security then an error message will be returned.

##### [B] Schedule

Required for version 1.3

**[C] URL Scheme**

```
DELETE /product-security/products/<product-id>
```

**[D] Examples**

Client request:

```
DELETE /product-security/products/72
```

Server response:

```
HTTP/1.1 204 No Content
```

```
Status: 204
```

```
X-API-Version: 1.3
```

**4.16.6 Remove Group from Product****[A] Description**

This API is used to remove the group from existing Product Security Product. If user does not have privileges to manage Product Security then an error message will be returned.

**[B] Schedule**

Required for version 1.3

**[C] URL Scheme**

```
DELETE /product-security/products/<product-id>/groups
```

**[D] Request Attributes**

Array which contains the group names.

**[E] Example**

Client request:

```
DELETE /product-security/products/72/groups
```

```
[
```

```
    "radar-automation",
```

```
    "Radar Access Group"
```

```
]
```

Server response:

```
HTTP/1.1 204 No Content
```

```
Status: 204
```

```
X-API-Version: 1.3
```



## 4.17 Product Security Release Vehicle

### 4.17.1 Get Release Vehicles List

#### [A] Description

This API is used to get all the Product Security Management Release Vehicles. If user do not have privileges to manage Product Security then an error message will be returned.

#### [B] Schedule

Required for version 1.2

#### [C] URL Scheme

```
GET /product-security/release-vehicles
```

#### [D] Response Attributes

The Response will be array of objects containing below attributes. If no Release Vehicle has been defined then an empty array ( [ ] )will be returned.

Key	Description	Data Type
id	ID of the release vehicle	Integer
name	Name of the Release Vehicle	String
isClosed	Is Release Vehicle closed.?	Boolean

#### [E] Examples

Client request:

```
GET /product-security/release-vehicles
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[ {
  "id": 231,
  "name": "iWork (iOS)",
  "isClosed": false
},
{
  "id": 232,
  "name": "iTunes Stores",
  "isClosed": false
},
... more records
```



1

#### 4.17.2 Create Release Vehicle

##### [A] Description

This API is used to create a new Product Security Release Vehicle. If user does not have privileges to manage Product Security then an error message will be returned.

##### [B] Schedule

Required for version 1.3

##### [C] URL Scheme

POST /product-security/release-vehicles

##### [D] Request Attributes

Key	Description	Data Type
name	Name of the release vehicle	String
isClosed	Is release vehicle closed?	Boolean
isRestricted	Is the release vehicle restricted?	Boolean
groupNames	Access or Work groups to restrict access to members of the added groups	Array of Strings

##### [E] Response Attributes

Created release vehicle object will be returned with response.

Key	Description	Data Type
id	ID of the release vehicle	Integer
name	Name of the release vehicle	String
isClosed	Is release vehicle closed?	Boolean
isRestricted	Is the release vehicle restricted?	Boolean
groups	Access or Work groups to restrict access to members of the added groups. See <a href="#">Product Security Group</a> table for description.	Array of group object



**[E] Example**

Client request:

```
POST /product-security/release-vehicles
{
  "name": "iWork (Mac)",
  "isClosed": false,
  "isRestricted": true,
  "groupNames": [
    "radar-automation"
  ]
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.3
{
  "id": 72,
  "isRestricted": true,
  "name": "iWork (Mac)",
  "isClosed": false,
  "groups": [
    {
      "id": 19790,
      "name": "radar-automation",
      "type": "Access"
    }
  ]
}
```

**4.17.3 Open Release Vehicle****[A] Description**

This API is used to open a specified Product Security Release Vehicle. If user does not have privileges to manage Product Security then an error message will be returned.

**[B] Schedule**

Required for version 1.3

**[C] URL Scheme**

```
GET /product-security/release-vehicles/<release-vehicle-id>
```

**[D] Response Attributes**

Response attributes are same as described in [Response Attributes](#) of section 4.17.2 Create a Product Security Release Vehicle.

**[E] Example**

Client request:

```
GET /product-security/release-vehicles/72
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.3
{
  "id": 72,
  "isRestricted": true,
  "name": "iPhone",
  "isClosed": false,
  "groups": [
    {
      "id": 19790,
      "name": "Phone Orpheus",
      "type": "Access"
    },
    {
      "id": 637,
      "name": "Test Casers",
      "type": "Work"
    }
  ]
}
```

**4.17.4 Update Release Vehicle****[A] Description**

This API is used to update an existing Product Security Release Vehicle. If user does not have privileges to manage Product Security then an error message will be returned.

**[B] Schedule**

Required for version 1.3

**[C] URL Scheme**

```
PUT /product-security/release-vehicles/<release-vehicle-id>
```

**[D] Request Attributes**

Request attributes are same as described in [Request Attributes](#) of section 4.17.2 Create a Product Security Release Vehicle.

**[E] Example**



Client request:

```
PUT /product-security/release-vehicles/72
{
  "name": "iWork (Mac)",
  "isClosed": false,
  "isRestricted": true,
  "groupNames": [
    "radar-automation"
  ]
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.3
```

#### 4.17.5 Delete Release Vehicle

##### [A] Description

This API is used to delete a Product Security Release Vehicle. If user does not have privileges to manage Product Security then an error message will be returned.

##### [B] Schedule

Required for version 1.3

##### [C] URL Scheme

```
DELETE /product-security/release-vehicles/<release-vehicle-id>
```

##### [D] Examples

Client request:

```
DELETE /product-security/release-vehicles/72
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.3
```



#### 4.17.6 Remove Group from Release Vehicle

##### [A] Description

This API is used to remove the group from existing Product Security Release Vehicle. If user does not have privileges to manage Product Security then an error message will be returned.

##### [B] Schedule

Required for version 1.3

##### [C] URL Scheme

```
DELETE /product-security/release-vehicles/<release-vehicle-id>/groups
```

##### [D] Request Attributes

Array which contains the group names.

##### [E] Example

Client request:

```
DELETE /product-security/release-vehicles/72/groups
[
  "radar-automation",
  "Radar Access Group"
]
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.3
```

### 4.18 SSP (SOAP/XML) based services

#### 4.18.1 Validate Radar ID

##### [A] Description

The functionality of this API is to validate the radar ID passed in request and return the title and state of the problem.

This function can be achieved using GetProblemByID. The GetProblemByID will take problemID as request and return default fields of response mentioned under ERS for WebServices 1.0. It will also check whether logged-in user has privileges to view the particular problem.

Response can be restricted to return only specified field using X-Fields-Requested header.

**[B] Examples**

Client request:

```
GET /problems/300000
X-Fields-Requested: title,state
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
{
  "title": "Radar Title of the Problem",
  "state": "Analyze"
}
```

**4.18.2 Update Radar Problem****[A] Description**

The functionality of this API is to update the problem diagnosis and upload a file to a problem.

The function of appending a diagnosis can be achieved using AppendToProblemDiagnosis API of Web Service 1.0. This API will take a problemID and diagnosis text as an input.

The function of uploading a file in problem can be achieved using UploadEnclosure API.

**[B] Examples****Append problem diagnosis**

Client request:

```
POST /problems/9000000/diagnosis
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "text": "There's chocolate in my peanut butter!"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

**Add an attachment**

Client request:

```
PUT /problems/9000000/attachments/Movies/IMG_0004.MOV
X-API-Version: 1.0
Content-Type: video/quicktime
<Binary Attachment Content>
```



Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### 4.18.3 Get Scheduled Test

##### [A] Description

The Functionality of this API is used to find the scheduled test based on search criteria mentioned in request.

This functionality can be achieve using FindScheduledTest API of Web Services 1.0.

#### 4.18.4 Fetch Problem Details

##### [A] Description

The functionality of this API is used to search all the problem which satisfy the mentioned search criteria passed in request.

This functionality can achieve using FindProblem API of WebServices 1.0

#### 4.18.5 Related Sonar

##### [A] Description

The client will send the radar Id, keyword Id, originator DS Id and Sonar Request Id. The web service checks the relation between the sonar Id and the external system and will return back the related sonar request if a single sonar Id is related to the external system. In other scenarios the web service will return back the state as 0 or 1

Business Logic:-

- The client passes the radar Id, keyword Id, originator DS Id and Sonar Request Id to the web service.
- Web service checks whether the problem originator is same as the input originator Id. If not same it returns state as 1. If they are same it proceeds to the next step.
- It will check whether the keyword Id is related to the input problem Id. If not related then returned the state as 0 and exits.
- Web service checks the related items in the external system for the radar. When no Sonar is related then it will return state as 0.
- When a single sonar request Id is related, then the return value will have the related Sonar Id.
- When multiple sonar Ids are related then it will return the state as 0.
- The return values will be send to the client via the web service response.
- If the relating sonar is not success an error message will be returned to the client.

**[B] Schedule**

Required for version 1.2

**[C] URL Scheme**

POST problems/<problemID>/relateSonar

**[D] Request Attributes**

The Request attributes will contain the below attributes and all the attributes are mandatory

Key	Description	Data Type
keywordID	ID of the keyword	Integer
originatorID	DSID of the originator of problem	Integer
sonarID	ID of the related sonar	Integer

**[D] Response Attributes**

The response attributes will contain the below attributes.

Key	Description	Data Type
state	state of the related sonar	Integer

Client request:

```
POST /problems/3000000/relateSonar
{
  "keywordID":28719,
  "originatorID":8794,
  "sonarID": 718976
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
{"state":0}
```

**4.18.6 Get Problem Statistics****[A] Description**



Problem Statistics aggregates the count of various radar problems between specific duration, specified in the request, and provides count for specific components , priorities, milestones etc on the days in the given duration.

For all the days from start date to end date, the statistics value will be returned back. If any of the days do not have statics data, then the previous date statistics values will be filled in the missing days.

When the user wants to calculate the statistical data without grouping by the dates, they can send the frequency as NonDate.

#### [B] Schedule

Required for version 1.2

#### [C] URL Scheme

POST `problems/problemStatistics`

#### [D] Request Attributes

Request Object will contain the below mentioned fields. All the Mandatory fields should be present in request object.

Key	Description	Data type	Mandatory
dateRange	The date range for which the statistics should be calculated Please refer Date Object given below for format.	Date Object	Yes
filterConditions	The conditions based on which the problems should be filtered for statistics calculation Please refer Filter Object given below for format.	Filter Object	Yes
frequency	If frequency is 'NO' the statistics is calculated taking data from start date to end date and the grouping will happen based on the pic columns and the date.  If the frequency is set as 'NonDate', then the data is taken from start date to end date and statistics is calculated without grouping by the date.  If the frequency is set as 'CURRENT', then date range is not needed. The statistics is calculated taking all the records into consideration.  The default value for frequency is 'NO'.	String	No





Key	Description	Data type	Mandatory
pickColumns	The statistics is counted by grouping the problems based on the pick columns that are send in the request. One or more pick columns will group the problems based on all the pick columns in the request. At least one pick column should be available in the request. Please refer PickColumn Object given below for format.	PickColumn Object	YES

### Date Object

Key	Description	Data type	Mandatory
startDate	The date from which the statistics value should be calculated	Date String	Yes
endDate	The date until which the statistics value should be calculated. If the endDate is not specified, current date is taken as the end date.	Date String	No

### Filter Object

Key	Description	Data type	Mandatory
componentName	The String array of component names	String array	Yes
componentVersion	The String array of component versions	String array	No
state	The String array of state names	String array	No
priority	The String array of priority code	String array	No
IncludeSubComponents	If send as 'YES', The statistics will include problems from the sub components also.	String array	No
substate	The String array of substate names	String array	No
resolution	The String array of resolution names	String array	No
classification	String array of classification names	String array	No
reproducible	String array of reproducibility title	String array	No
milestone	String array of milestone build version	String array	No
taskOrder	String array of task order	String array	No



Key	Description	Data type	Mandatory
fixOrder	String array of fix ordering codes	String array	No
keyword	The keywords that the problem has	String array	No

**PickColumn object**

Key	Description	Data type	Mandatory
priority	Set as 'YES' to group the results by priority.	String	No
state	Set as 'YES' to group the results by state.	String	No
milestone	Set as 'YES' to group the results by milestone.	String	No
fixOrder	Set as 'YES' to group the results by fixorder.	String	No
component	Set as 'YES' to group the results by component.	String	No
resolution	Set as 'YES' to group the results by resolution.	String	No

**[E] Response Attributes**

Response will be array of below mentioned fields

Key	Description	Data Type
date	The date in for which the statistical data is taken	String
priority	The priority code for which the problems were grouped	String
state	The state name for which the problems were grouped	String
milestone	The milestone build version for which the problems were grouped	String
fixOrder	The fix order for which the problems were grouped	String
component	The component name for which the problems were grouped	String
resolution	The resolution name for which the problems were grouped	String



Key	Description	Data Type
count	The number of problems belonging to the above mentioned categories.	Integer

**[E] Example**

Client request:

```
POST /problems/problemStatistics
{
  "dateRange": {
    "startDate": "21-Sep-2012",
    "endDate": "30-Dec-2012",
  },
  "filterConditions": {
    "componentName": [
      "radar"
    ],
    "componentVersion": [
      "7.1"
    ],
    "includeSubComponents": "YES"
  },
  "frequency": "YES",
  "pickColumns": {
    "priority": "YES",
    "state": "YES",
    "milestone": "YES",
    "fixOrder": "YES",
  }
}
```

Server response:

HTTP/1.1 201 Created

X-API-Version: 1.0

```
[
  {
    "milestone": "7.1-pre-release change",
    "count": 1,
    "fixOrder": 6,
    "priority": 5,
    "state": "Closed",
    "date": "21-SEP-2012"
  },
  {
    "milestone": "7.1-pre-release change",
    "count": 7,
    "fixOrder": 6,
    "priority": 3,
```



```

        "state": "Closed",
        "date": "21-SEP-2012"
    },
    ... more records
]
Client request:

POST /problems/problemStatistics
{
    "dateRange": {
        "startDate": "21-Sep-2012",
        "endDate": "30-Dec-2012",

    },
    "filterConditions": {
        "componentName": [
            "radarweb"
        ],
        "includeSubComponents": "YES"
    },
    "frequency": "YES",
    "pickColumns": {
        "priority": "YES",
        "state": "YES",
        "milestone": "YES",
        "fixOrder": "YES",

    }
}
Server response:
HTTP/1.1 201 Created
X-API-Version: 1.0
[
    {
        "count": 3,
        "fixOrder": 6,
        "priority": 5,
        "state": "Analyze",
        "date": "21-SEP-2012"
    },
    ... more records
]

```

#### 4.19 Related Tests

This set of APIs deals with relating “TSTT” to a given Radar problem.



#### 4.19.1 Get Related Tests

##### [A] Description

This API provides a method to retrieve the list of related tests items. Empty array will be returned if there is no related tests in the specified problem.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
GET /problems/<problem_id>/related-tests
```

##### [D] Response Attributes

Key	Description	Data Type
id	ID of the Problem	Integer
title	Title of the Problem	String
testType	One of "Scheduled Test", "Scheduled Test Case", "Test Suite" and "Test Suite Case".	String
relationType	Relation type of Problem.	String
isResolved	It will be a boolean value.	Boolean

##### [E] Examples

**Retrieve the list of related tests:**

Client request:

```
GET /problems/900000/related-tests
X-API-Version: 1.0
```

Server response:

```
[
  {
    "id": 1114,
    "title": "areTesting for Rada WS -1.3",
    "testType": "Scheduled Test",
    "relationType": "blocking",
    "isResolved": true
  }
]
```



```

    },
    {
      "id": 1114,
      "title": "areTesting for Rada WS -1.3",
      "testType": "Scheduled Test",
      "relationType": "related to",
      "isResolved": true
    }
  ]

```

#### 4.19.2 Edit Related Tests

##### [A] Description

This API provides a method to edit existing related test of problem. On success, the server responds with 201 Created and no response body.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

POST /problems/<problem\_id>/related-tests

##### [D] Request Attributes

Key	Description	Data Type
relationType	One of: "related-to", "blocked-by", "blocking".	String
testType	One of "scheduled-test", "scheduled-test-case", "test-suite" and "test-suite-case".	String
oldRelationType	One of: "related-to", "blocked-by", "blocking".	String
testID	ID of the related tests	Integer

##### [E] Examples

###### Edit related test:

```

POST /problems/900000/related-tests
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8

```



```
{
  "relationType": "related-to",
  "testType": "test-suite",
  "oldRelationType" : "blocked-by",
  "testID" : 280043
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

### 4.19.3 Add Related Tests

#### [A] Description

This API provides a method to add a relation to the list of related tests. The request consists solely of a PUT and a URL path, without a request body. On success, the server responds with 201 Created and no response body.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
PUT /problems/<problem-id>/related-tests/<relation-type>/<test-type>/<
related-test-id>
```

#### [D] Request Attributes

The request has no body. The relation\_type in the url can be one of: "related-to", "blocked-by", "blocking" and test\_type can be one of "scheduled-test", "scheduled-test-case", "test-suite" and "test-suite-case".

#### [E] Examples

Client request:

```
PUT /problems/900000/related-tests/blocking/scheduled-test/1315306
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 201 Created
X-API-Version: 1.0
```



#### 4.19.4 Remove Related Tests

##### [A] Description

This API used to remove related test from specified problem ID. On Success, server will responds with 204 No Content and no response body.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

```
DELETE /problems/<problem-id>/related-tests/<relation-type>/<test-type>/  
<related-test-id>
```

##### [D] Request Attributes

The request has no body. The relation\_type in the url can be one of: "related-to", "blocked-by", "blocking" and test\_type can be one of "scheduled-test", "scheduled-test-case", "test-suite" and "test-suite-case".

##### [E] Examples

Client request:

```
DELETE /problems/12337883/related-tests/related-to/scheduled-test/1114  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content  
Status: 204  
X-API-Version: 1.0
```

### 4.20 Watched Problems

This set of APIs deals with adding and removing from Watched problems list of logged-in user.

#### 4.20.1 Add Problem to Watched Problem List

##### [A] Description

This API is used to add problems to the watched problem list of the logged-in person. The request consists solely of a PUT and a URL path, without a request body. On success, the server responds with 201 Created and no response body.

An attempt made to add problems already part of the watch list will result in a successful response but no change will be made.

A validation will be done on problem existence and accessibility based on logged-in user and appropriate error message will be thrown on failure.



**[B] Schedule**

Required for version 1.5

**[C] URL Scheme**

PUT /people/find/current-user/watched-problems/<problem-id>

**[D] Examples****Add problem in my watched problem list**

Client request:

PUT /people/find/current-user/watched-problems/4000000

Server response:

HTTP/1.1 201 Created  
Status: 201

**Add already added problem in my watched problem list**

Client request:

PUT /people/find/current-user/watched-problems/4000000

Server response:

HTTP/1.1 400 Bad Request  
Content-Type: application/json; charset=utf-8  
{  
 "message": "Problem ID '4000000' is already added to Watched Problem List.",  
 "title": "Duplicate Watched Problem",  
 "help": "View documentation at <http://radar.apple.com/>",  
 "status": "400 Bad Request"  
}

**4.20.2 Remove Problem from Watched Problem List****[A] Description**

This API is used to remove problems from watched problems list of logged-in person. The request consists solely of a DELETE and a URL path, without a request body. On success, the server responds with 204 No Content and no response body.

A validation will be done on problem existence and accessibility based on logged-in user and appropriate error message will be thrown on failure.

**[B] Schedule**

Required for version 1.5

**[C] URL Scheme**

```
DELETE /people/find/current-user/watched-problems/<problem_id>
```

**[D] Examples****Remove problem from Watched Problem List:**

Client request:

```
DELETE /people/find/current-user/watched-problems/9000000
```

Server response:

```
HTTP/1.1 204 No Content  
Status: 204
```

## 4.21 Get Configuration Text

**[A] Description**

This API provides a method to retrieve configuration text based on configuration ID. If configuration ID is not a valid ID then an error message will be shown.

**[B] Schedule**

Required for version 1.3

**[C] URL Scheme**

```
GET /configurations/<config-ID>
```

**[D] Response Attributes**

Response will be an array of relationship tree object. Each object will contain the below mentioned attributes.

Key	Description	Data Type
id	ID of the configuration	Integer
name	Title of the configuration	String
isDefault	Is configuration text default for problem	Boolean



Key	Description	Data Type
text	Configuration text	String

### [E] Examples

Client request:

```
GET /configurations/48910
X-API-Version: 1.3
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: application/json; charset=utf-8
{
  "id": 48910,
  "name": "Mac Configuration",
  "isDefault": true,
  "text": "%B%M Mac Book"}
```

## 4.22 Query APIs

### 4.22.1 Get Recent Query List

#### [A] Description

Fetches the 20 most recently executed queries for the logged-in user. If logged-in user does not have any recent queries then empty array will be returned in response. Queries executed for automation purposes are not added to the recent queries list. To best manage the list, only queries executed by Radar clients are retained as recent queries.

This API is optimized for OS X and iOS Clients.

#### [B] Schedule

Required for version 1.6.0

#### [C] URL Scheme

```
GET /queries/recent
```

#### [D] Response Attributes

Response will contain array of below attribute



Key	Description	Data Type
recentID	Primary Key of Recent Query	Integer
name	Client-generated name to describe the Query	String
type	Type of objects on which query searches.	Enumerated String
lastModifiedAt	Last modification or executed date of Recent Query	ISO 8601 Date Time String
executedAs	Identifies what the user was doing when the query was saved to recents	"Ad-Hoc" OR "Saved"

### [E] Example

Client request:

```
GET /queries/recents
```

Server response:

HTTP/1.1 200 OK

```
{
  "status":{
    "code":"Success",
    "message":"3 recent queries returned."
  },
  "recents":[
    {
      "name": "CC'dCount=123654879",
      "recentID": 665152,
      "type": "Problem",
      "lastModifiedAt": "2014-06-04T09:14:23+0000",
      "executedAs": "Ad-Hoc"
    },
    {
      "name": "my daily query",
      "recentID": 663431,
      "type": "Problem",
      "lastModifiedAt": "2014-06-04T09:14:23+0000",
      "executedAs": "Ad-Hoc"
    },
    {
      "name": "saying grace",
      "recentID": 665103,
      "type": "TSTT",
      "lastModifiedAt": "2014-06-04T09:14:23+0000",
      "executedAs": "Saved"
    }
  ]
}
```



#### 4.22.2 Get Individual Recently Executed Query

##### [A] Description

This API will fetch a single query executed by logged-in user.

This will be a Private API used by OS X and iOS Clients.

##### [B] Schedule

Required for version 1.6.0

##### [C] URL Scheme

```
GET /queries/recent/<recent-id>
```

##### [D] Response Attributes

Response will contain array of below attribute

Key	Description	Data Type
recentID	Primary Key of Recent Query	Integer
name	Client-generated name to describe the Query	String
type	Type of objects on which query searches.	Enumerated String
lastModifiedAt	Last modification or executed date of Recent Query	ISO 8601 Date Time String
executedAs	Identifies what the user was doing when the query was saved to recent	"Ad-Hoc" & "Saved"

##### [E] Example

Client request:

```
GET /queries/recent/1002
```

Server response:

```
HTTP/1.1 200 OK
{
  "status": {
    "code": "Success",
```



```

    "message": "1 recent queries returned."
  },
  "recent": {
    "name": "CC'dCount=123654879",
    "recentID": 665152,
    "type": "Problem",
    "lastModifiedAt": "2014-06-04T09:14:23+0000",
    "executedAs": "Ad-Hoc"
  },
}
}

```

#### 4.22.3 Execute Recent Query API

##### [A] Description

Fetches results of a single recently executed query by its id. Execute Recent Query response can be of type Find Problem, Find Test Suite or Find Scheduled Tests unless countsOnly is set to true in the initial search criteria. Queries with idsOnly set to true return the full Problem or TSTT objects in results for recent queries. If recent query id is of type not supported by RWS then exception message will be thrown.

##### [B] Schedule

Required for version 1.6.0

##### [C] URL Scheme

```
GET /queries/recent/<recent-id>/results
```

##### [D] Response Attributes

Similar to Find Problem Response

Queries With "countsOnly" as true

Key	Description	Data Type
totalCount	count of find results	Integer

##### [E] Example

Client request:

```
GET /queries/recent/1002/results
```

Server response:



HTTP/1.1 200 OK

```
{
  "status":{
    "code":"Success",
    "message":"2 objects returned."
  },
  "results":[
    {
      "milestone": null,
      "substate": null,
      "id": 1,
      "fingerprint": "9eeb170d",
      "title": "AC106",
      "component": {
        "name": "Snickerdoodle",
        "version": "1.1"
      },
      "fixOrder": 1,
      "priority": 3,
      "classification": "Crash/Hang/Data Loss",
      "state": "Closed",
      "assignee": {
        "lastName": "Miller",
        "email": null,
        "type": null,
        "firstName": "Howard",
        "dsid": 6760
      },
      "lastModifiedAt": "2013-10-22T11:16:49+0530"
    },
    {
      "milestone": null,
      "substate": null,
      "id": 111111,
      "fingerprint": "dc830d37",
      "title": "(a3 Installer 3.0) Print Monitor and Utility paths
created when not installed.",
      "component": {
        "name": "Win Installer Script",
        "version": "1.0"
      },
      "fixOrder": 2,
      "priority": 5,
      "classification": "Serious Bug",
      "state": "Closed",
      "assignee": {
        "lastName": "Foreman",
        "email": null,
        "type": null,
        "firstName": "Shirley",
        "dsid": 11667
      },
      "lastModifiedAt": "2013-10-22T07:02:12+0530"
    }
  ]
}
```



```
]
}
```

#### Query With "countsOnly" as true

Initial Find Criteria:

```
{ "state": "Closed", "component": { "name": "Radar", "version": "Automation" },
  "countsOnly": true }
```

Client request:

```
GET /queries/recent/12355/results
```

Server response:

```
HTTP/1.1 200 OK
```

```
{
  "status": {
    "message": "1 objects returned.",
    "code": "Success"
  },
  "results": [
    { "totalCount": 348 }
  ]
}
```

### 4.22.4 Execute Shared Report

#### [A] Description

This API provides a method to execute shared report. It accepts reportID from client and execute it. This API will only support 'Problem Query' type of shared report.

If logged-in user does not have access to passed report then an error message will be shown. An error message will also be shown if report type is other than 'Problem Query'.

Response attribute returned after shared report execution will be same as Find Problem API mentioned in sec 3.2. User can also request non-default fields by using X-Fields-Requested header.

The default number of rows returned is 2000 and is also the maximum number of rows that can be returned. User can change the number of rows returned by using the "X-rowlimit" header.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

```
GET shared-reports/<reportID>
```





## [D] Response Attributes

Similar to Find Problem Response.

## [E] Examples

Client request:

```
GET /shared-reports/123595
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: application/json; charset=utf-8
```

```
[
  {
    "milestone": null,
    "substate": null,
    "id": 1,
    "fingerprint": "9eeb170d",
    "title": "AC106",
    "component": {
      "name": "Snickerdoodle",
      "version": "1.1"
    },
    "fixOrder": 1,
    "priority": 3,
    "classification": "Crash/Hang/Data Loss",
    "state": "Closed",
    "assignee": {
      "lastName": "Miller",
      "email": null,
      "type": null,
      "firstName": "Howard",
      "dsid": 6760
    },
    "lastModifiedAt": "2013-10-22T11:16:49+0530"
  },
  {
    "milestone": null,
    "substate": null,
    "id": 111111,
    "fingerprint": "dc830d37",
    "title": "(a3 Installer 3.0) Print Monitor and Utility paths
created when not installed.",
    "component": {
      "name": "Win Installer Script",
      "version": "1.0"
    },
    "fixOrder": 2,
    "priority": 5,
```



```

"classification": "Serious Bug",
"state": "Closed",
"assignee": {
  "lastName": "Foreman",
  "email": null,
  "type": null,
  "firstName": "Shirley",
  "dsid": 11667
},
"lastModifiedAt": "2013-10-22T07:02:12+0530"
},
{
  "milestone": {
    "component": {
      "name": "Radar",
      "version": "7.3"
    },
    "name": "TBD"
  },
  "substate": "Screen",
  "id": 2003675,
  "fingerprint": "93943771",
  "title": "Validate person -> validate only those for the component
assigned",
  "component": {
    "name": "Radar",
    "version": "7.3"
  },
  "fixOrder": 1,
  "priority": 4,
  "classification": "UI/Usability",
  "state": "Analyze",
  "assignee": {
    "lastName": "Manavi",
    "email": null,
    "type": null,
    "firstName": "Marjan",
    "dsid": 1288017191
  },
  "lastModifiedAt": "2013-10-22T10:11:12+0530"
},
{
  "milestone": {
    "component": {
      "name": "Radar",
      "version": "Future"
    },
    "name": "TBD"
  },
  "substate": null,
  "id": 2296300,
  "fingerprint": "eba3c554",
  "title": "It would be nice if the whole Problem Diagnosis 'row'
acted as resize",
  "component": {

```



```
        "name": "Radar",
        "version": "Future"
    },
    "fixOrder": 6,
    "priority": 4,
    "classification": "Enhancement",
    "state": "Verify",
    "assignee": {
        "lastName": "System",
        "email": null,
        "type": null,
        "firstName": "iQA_Automation",
        "dsid": 1770812598
    },
    "lastModifiedAt": "2013-10-22T06:15:25+0530"
},
{
    "milestone": {
        "component": {
            "name": "Radar",
            "version": "7.3"
        },
        "name": "TBD"
    },
    "substate": null,
    "id": 2327059,
    "fingerprint": "fe9123fd",
    "title": "Unique the Find Problem window",
    "component": {
        "name": "Radar",
        "version": "7.3"
    },
    "fixOrder": 6,
    "priority": 4,
    "classification": "Power",
    "state": "Verify",
    "assignee": {
        "lastName": "Tester1",
        "email": null,
        "type": null,
        "firstName": "Radar",
        "dsid": 1118580968
    },
    "lastModifiedAt": "2013-10-31T04:50:24+0530"
},
{
    "milestone": {
        "component": {
            "name": "Radar",
            "version": "7.3"
        },
        "name": "TBD"
    },
    "substate": null,
    "id": 2336171,
```



```

    "fingerprint": "37b34d6c",
    "title": "\"Save Form Setup\" for the Find Problem Window",
    "component": {
        "name": "Radar",
        "version": "7.3"
    },
    "fixOrder": 6,
    "priority": 4,
    "classification": "Power",
    "state": "Integrate",
    "assignee": {
        "lastName": "Developer",
        "email": null,
        "type": null,
        "firstName": "Radar",
        "dsid": 568400
    },
    "lastModifiedAt": "2013-10-28T10:43:27+0530"
},
{
    "milestone": {
        "component": {
            "name": "Auto iQA Root Component",
            "version": "7.3"
        },
        "name": "AutoiQAMilestone"
    },
    "substate": null,
    "id": 2383032,
    "fingerprint": "e9d4fbf9",
    "title": "Test New Problem Title0.735873455349",
    "component": {
        "name": "Auto iQA Root Component",
        "version": "7.3"
    },
    "fixOrder": 2,
    "priority": 5,
    "classification": "UI/Usability",
    "state": "Verify",
    "assignee": {
        "lastName": "Carmichael",
        "email": null,
        "type": null,
        "firstName": "Derek",
        "dsid": 119268
    },
    "lastModifiedAt": "2013-10-30T16:45:38+0530"
}]

```



#### 4.22.5 Get Shared Reports

##### [A] Description

This API will be used to get all shared report which logged-in user has access. It will return shared reports which are active and public and accessible by logged-in user.

##### [B] Schedule

Required for version 1.4

##### [C] URL Scheme

GET /shared-reports

##### [D] Response Parameters

Parameter	Description	Data Type
id	ID of Shared Report	Integer
name	Name of Shared Report	String
description	Description of Shared Report	String
type	Report Type of Shared Report. Enumeration for type are mentioned in <a href="#">2.8.2. Get Shared Report Type Enumerations</a>	Enumerated Value
isPublic	Is Shared Report publicly executable.?	Boolean
isActive	Is Shared Report active.?	Boolean
owner	Owner of Shared Report	Person Object
createdAt	Creation date of Shared Report	ISO 8601 Datetime string
lastModifiedBy	Person last modified the shared report	Person Object
lastModifiedAt	Last modification date of shared report	ISO 8601 Datetime string

##### [E] Examples

Client request:

GET /shared-reports

Server response:

HTTP/1.1 200 OK



```
Status: 200
Content-Type: application/json; charset=utf-8

[
  {
    "id": 127474,
    "name": "!4.1 Open problem",
    "description": null,
    "reportType": "Problem Query",
    "isPublic": true,
    "isActive": true,
    "createdAt": "2013-11-05T21:31:34+0000",
    "lastModifiedAt": "2013-11-05T21:31:34+0000",
    "owner": {
      "dsid": 270256630,
      "firstName": "Patrick",
      "lastName": "Hong",
      "email": null,
      "type": null
    },
    "lastModifiedBy": {
      "dsid": 270256630,
      "firstName": "Patrick",
      "lastName": "Hong",
      "email": null,
      "type": null
    }
  },
  ...more records
]
```

#### 4.22.6 Create Shared Report

##### [A] Description

This API will create a new shared report based on search criteria passed in request. Attribute 'name' and 'searchAttributes' are mandatory in request to create a new report. The reportID passed in response can later be used to execute using existing Execute Shared Report API.

RWS will only support creation of Shared Report of type Find Problem Advanced with reserved extension 'rwsFindProbAdv15' for RWS.

Attribute 'searchAttributes' should contain the same json data which is used as Find Problem search criteria. The Json data passed in 'searchAttributes' will be validated as per Find Problem API rule and if it has some invalid data then as error message will be shown.

The report name uniqueness will also be validated to ensure that there will not be any duplicate shared report with same name for same owner.

##### [B] Schedule



Required for version 1.6.0

### [C] URL Scheme

POST /shared-reports

### [C] Request Attributes

Key	Description	Data Type
name	Name of shared report. Max supported length is 1024 char.	String
description	Description of shared report. Mac supported length is 4000 char.	String
isPublic	Flag to indicate shared report public	Boolean
searchAttributes	A Find search criteria in json object for which shared report need to created.	Object
type	Report Type of Shared Report	Enumerated Value

### [D] Response Attributes

Key	Description	Data Type
id	ID of shared report.	Integer
name	Name of shared report. Max supported length is 1024 char.	String
description	Description of shared report. Max supported length is 4000 char.	String
isPublic	Flag to indicate shared report public	Boolean
isActive	Flag to indicate shared report active	Boolean
createdAt	Creation date of shared report	ISO 8601 Date String
lastModifiedAt	Last modification date of shared report	ISO 8601 Date String
type	Report Type of Shared Report	Shared Reports

### [E] Example



Client request:

```
POST /shared-reports
{
  "name": "Problems under Radar|Automation",
  "description": "Shared report to return all the problem associated
to component Radar | Automation",
  "isPublic": true,
  "searchAttributes":
    {
      "component":{
        "name":"Radar",
        "version":"Automation"
      }
    }
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
Content-Type: application/json; charset=utf-8
{
  "id": 133123,
  "name": "Problems under Radar|Automation",
  "description": "Shared report to return all the problem associated
to component Radar | Automation",
  "isPublic": true,
  "isActive": true,
  "created": "2014-03-20T12:16:49+0000",
  "lastModifiedAt": "2014-03-20T12:16:49+0000"
}
```

### Shared Report Name Uniqueness Error

Client Request

```
POST /shared-reports
{
  "name": "Problems under Radar|Automation",
  "description": "Shared report to return all the problem associated
to component Radar | Automation",
  "isPublic": true,
  "searchAttributes":
    {
      "component":{
        "name":"Radar",
        "version":"Automation"
      }
    }
}
```





Server response:

```
HTTP/1.1 400 Bad Request
Status: 400
Content-Type: application/json; charset=utf-8
{
  "message": "A Shared Report with name 'Problems under Radar|
Automation' is already created. Please choose some other report name.",
  "help": "View documentation at http://radar.apple.com/",
  "title": "Invalid Request",
  "status": "400 Bad Request"
}
```

#### 4.22.7 Get Shared Report Subscriber List

##### [A] Description

This API provides a method to retrieve the list of subscriber of the shared report.

##### [B] Schedule

Required for version 1.6.0

##### [C] URL Scheme

```
GET /shared-reports/<report_id>/subscribers
```

##### [D] Response Attributes

Response will contain an array of Subscriber object mentioned in Section 4.19.7 Get Query Details API

##### [E] Example

Client request:

```
GET /shared-reports/21141/subscribers
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "permission": "Read Only",
    "type": "Access Group",
    "subscriber": {
      "id": 9599,
      "name": "Radar - Reassignable Folks"
    }
  },
  {

```



```
"permission": "Read & Write",
"type": "Work Group",
"subscriber": {
  "id": 8650,
  "name": "Radar"
},
{
  "permission": "Owner (Read & Write)",
  "type": "Person",
  "subscriber": {
    "lastName": "Tester1",
    "email": "radartester01@gmail.com",
    "type": "Contractor",
    "firstName": "Radar",
    "dsid": 382990196
  }
},
{
  "permission": "No Access",
  "type": "Everyone"
}
]
```

#### 4.22.8 Get Query Details API

##### [A] Description

Fetches full details of a single query by its id.

##### [B] Schedule

Required for version 1.6.0

##### [C] URL Scheme

```
GET /queries/<report-id>/detail
```

##### [D] Response Attributes

Key	Description	Data Type	Default
id	Id of Shared Report / Query	Integer	Y
name	Name of Shared Report / Query	String	Y
description	Description of Shared Report / Query	String or null	Y
isPublic	Flag to indicate whether Shared Report / Query is public	Boolean	Y



Key	Description	Data Type	Default
isActive	Flag to indicate whether Shared Report / Query is active	Boolean	Y
type	Type of Shared Report. Enumeration value can be found in section 2.10.2 Query / Shared Report type Enumeration	Enumerated value	Y
createdAt	Creation date of Shared Report / Query	ISO 8601 DateTime String	Y
createdBy	Person who created Shared Report / Query	Person Object	N
lastModifiedAt	Last modification date of Shared Report / Query	ISO 8601 DateTime String	Y
lastModifiedBy	Person who last modified Shared Report / Query	Person Object	N
subscribers	Person and groups subscribed for Shared Report / Query	Array of Subscribers Object	N

#### Subscribers Object

Key	Description	Data Type
type	Type of subscriber. Enumeration value can be found in section 2.10.3 Subscriber Type Enumeration	Enumerated value
permission	Type of subscriber. Enumeration value can be found in section 2.10.4 Subscriber Permission Enumeration	Enumerated value
subscriber	Subscriber details. If type is Person then will contain Person Object otherwise Group object. For type everyone, it will contain only name in object.	Person Object or Group Object

#### Person Object

Key	Description	Data Type
dsid	DSID of person	Integer
firstName	First name of person	String
lastName	Last name of person	String
email	Email of Person	String



Key	Description	Data Type
type	Type of Person	Enumerated String

**Group Object**

Key	Description	Data Type
id	ID of Group	Integer
name	Name of Group	String

**[E] Example****Get Shared Report Details with default attributes**

Client request:

GET /queries/1002/detail

HTTP/1.1 200 OK

```
{
  "status":{
    "code":"Success",
    "message":"3 recent queries returned."
  },
  "query":{
    "id":68997,
    "name":"Personal Test Suites",
    "description":"Personal Test Suites",
    "isPublic": false,
    "isActive": true,
    "type":"Test Suite List",
    "createdAt":"2014-04-05T13:03:00+0000",
    "createdBy":{
      "dsid":382990196,
      "firstName": "Radar",
      "lastName": "Tester1",
      "type":"Contractor",
      "email":"radartester01@gmail.com"
    },
    "lastModifiedAt":"2014-04-05T13:03:00+0000",
    "lastModifiedBy":{
      "dsid":382990196,
      "firstName": "Radar",
      "lastName": "Tester1",
      "type":"Contractor",
      "email":"radartester01@gmail.com"
    },
    "subscribers": [
      {
```



```

        "permission": "Read Only",
        "type": "Access Group",
        "subscriber": {
            "id": 9599,
            "name": "Radar - Reassignable Folks"
        }
    },
    {
        "permission": "Read & Write",
        "type": "Work Group",
        "subscriber": {
            "id": 8650,
            "name": "Radar"
        }
    },
    {
        "permission": "Owner (Read & Write)",
        "type": "Person",
        "subscriber": {
            "lastName": "Tester1",
            "email": "radartester01@gmail.com",
            "type": "Contractor",
            "firstName": "Radar",
            "dsid": 382990196
        }
    },
    {
        "permission": "No Access",
        "type": "Everyone"
    }
}

```

#### 4.22.9 Get Query Subscribed API

##### [A] Description

This API will be used to get all shared report which logged-in user has access.

If any of below mentioned first three criteria matches with mandatory fourth criteria then that report would be returned in response.

1. Logged-in person can be subscribed to shared report as member
2. Logged-in person can be member of group which is subscribed to shared report
3. Logged-in person is owner of report.
4. Shared Report should be active.

This Api will also support for X-rowlimit and X-Fields-Requested headers.If user will not set any X-rowlimit and X-Fields-Requested then default response would be 2000 rows with all response fields.

**[B] Schedule**

Required for version 1.6.0

**[C] URL Scheme**

GET /queries/subscribed

**[D] Response Parameters**

Parameter	Description	Data Type
id	ID of Shared Report	Integer
name	Name of Shared Report	String
description	Description of Shared Report	String
type	Report Type of Shared Report	Enumerated Value
isPublic	Is Shared Report publicly executable.?	Boolean
isActive	Is Shared Report active.?	Boolean
owner	Owner of Shared Report	Person Object
createdAt	Creation date of Shared Report	ISO 8601 Datetime string
lastModifiedBy	Person last modified the shared report	Person Object
lastModifiedAt	Last modification date of shared report	ISO 8601 Datetime string

**[E] Examples**

Client request:

```
GET /queries/subscribed
"X-Fields-Requested:id,isActive,createdAt,isPublic"
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8

{
  "status": {
    "message": "700 Objects returned.",
    "code": "Success"
  },
  "results": [
```



```

    {
      "id": 127474,
      "isActive": true,
      "createdAt": "2013-11-05T21:31:34+0530",
      "isPublic": true
    },
    {
      "id": 129567,
      "isActive": true,
      "createdAt": "2014-02-03T17:54:43+0530",
      "isPublic": true
    },
    {
      "id": 102750,
      "isActive": true,
      "createdAt": "2009-01-05T18:17:17+0530",
      "isPublic": true
    },
    {
      "id": 127028,
      "isActive": true,
      "createdAt": "2013-10-19T17:02:42+0530",
      "isPublic": true
    }
    .... more objects
  ]
}

```

#### 4.22.10 Add and Edit Subscribers in Shared Report

##### [A] Description

This API provides a method to add or edit a person or group in the subscriber list for a shared report. On success, the server responds with 201 Created and no response body.

If passed person or group not added as subscriber then this API will add it as subscriber and if it is already added then this API can be used to edit its privilege. If request contains same permission as existing permission in subscriber list for group or person then HTTP 400 Bad request with error message will be shown.

Only subscribers with permission as "Owner (Read & Write)" and "Read & Write" can add any person or group as subscriber or edit any other subscriber permission in shared report.

Permission "No Access" can be applied only for "everyone" and it cannot be applied to any other subscribed person or group. Subscriber "everyone" will support only two types of permission, "No Access" and "Read Only" and other will not be valid for it, if passed then proper error message will be shown.

Permission "Owner (Read & Write)" cannot be applied to any group, if applied then error message will be shown.



Permission for Subscriber having “Owner (Read & Write)” cannot be modified using this API.

Note:- Any group cannot be made as Owner of Shared Report using permission “Owner (Read & Write)”

#### [B] Schedule

Required for version 1.6.0

#### [C] URL Scheme

```
PUT /shared-reports/<report_id>/subscribers/people/<dsid>
PUT /shared-reports/<report_id>/subscribers/groups/access-groups/<group-name>
PUT /shared-reports/<report_id>/subscribers/groups/work-groups/<group-name>
PUT /shared-reports/<report_id>/subscribers/everyone
```

#### [D] Request Attributes

Key	Description	Data Type
permission	Permission on shared report for subscriber. Enumerated values are mentioned in Sec 2.3.4 Shared Report Subscriber Permission Enumeration.	Enumerated value

#### [E] Examples

**Add a person to the subscriber list:**

Client request:

```
PUT /shared-reports/21141/subscribers/people/8794
{
  "permission": "Read & Write"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
```

**Add an access group to the subscriber list:**

Client request:

```
PUT /shared-reports/21141/subscribers/groups/access-groups/RAM%20DAC%20ALL
```





```
{
  "permission": "Read Only"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
```

#### Error scenario with permission as No Access

Client request:

```
PUT /shared-reports/21141/subscribers/groups/access-groups/RAM%20DAC
%20ALL
```

```
{
  "permission": "No Access"
}
```

Server response:

```
HTTP/1.1 400 Bad Request
Content-Type: application/json; charset=utf-8
{
  "message": "Please pass only 'Read Only', 'Read & Write' or 'Owner (Read
& Write)' in permission parameter.",
  "title": "Invalid Request Object",
  "help": "View documentation at http://radar.apple.com/",
  "status": "400 Bad Request"
}
```

#### Error scenario with permission as Owner (Read & Write) for group

Client request:

```
PUT /shared-reports/21141/subscribers/groups/access-groups/RAM%20DAC
%20ALL
```

```
{
  "permission": "Owner (Read & Write)"
}
```

Server response:

```
HTTP/1.1 400 Bad Request
Content-Type: application/json; charset=utf-8
{
  "message": "You can not add a group as an owner of the shared report.",
  "title": "Can not add group as owner of the shared report",
}
```



```
"help": "View documentation at http://radar.apple.com/",  
"status": "400 Bad Request"  
}
```

#### 4.22.11 Remove Subscribers from Shared Report

##### [A] Description

This API provides a method to remove a person or group from the subscriber list of a shared report. The request consists solely of a DELETE and a URL path, without a request body. On success, the server responds with 204 No Content and no response body.

Subscriber person having "Owner (Read & Write)" permission and "everyone" cannot be removed using this API.

Only subscribers with permission as "Owner (Read & Write)" and "Read & Write" can remove any other subscriber from shared report.

##### [B] Schedule

Required for version 1.6

##### [C] URL Scheme

```
DELETE /shared-reports/<report_id>/subscribers/people/<dsid>  
DELETE /shared-reports/<report_id>/subscribers/groups/access-groups/  
    <group-name>  
DELETE /shared-reports/<report_id>/subscribers/groups/work-groups/  
    <group-name>
```

##### [D] Examples

###### Remove a person from the subscriber list:

Client request:

```
DELETE /shared-reports/12345/subscribers/people/8794
```

Server response:

```
HTTP/1.1 204 No Content  
Status: 204
```

###### Remove an access group from the subscriber list:

Client request:

```
DELETE /shared-reports/12345/groups/access-groups/RAM%20DAC%20A11
```



Server response:

```
HTTP/1.1 204 No Content
Status: 204
```

#### 4.22.12 Get Recently Opened Problems

##### [A] Description

This API is used to fetch the recent problems opened in detail view in a Radar client.

##### [B] URL Scheme

```
GET /problems/recent
```

##### [C] Response Attributes

Key	Description	Data Type
recent	List of recently opened Problems	Array of Problem Objects

##### Recent Problem Objects

Key	Description	Data Type
problem	Problem Included in Other Objects	Object
lastOpenedAt	Date and Time when the item was opened by the signed-in user	ISO 8601 Datetime string

##### Problems Included in Other Objects

Key	Description	Data Type
id	The problem ID.	Integer
assignee	The person who the problem is assigned to. See 7.2 People Included in Other Objects.	Object



component	The problem's component. See 6.5 Components Included in Other Objects.	Object
state	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String
title	The problem's title.	String
labels	Label details of the problem. See Response Attributes table in the section 11.1 Get Labels, for description.	Object

## [D] Examples

### Get Recent Problems

Client request:

```
GET /problems/recent
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.6
```

```
{
  "status": {
    "code": "Success",
    "message": "3 recent problems returned."
  },
  "recent": [
    {
      "lastOpenedAt": "2014-01-16T12:39:50+0000",
      "problem": {
        "id": 4000000,
        "title": "ER: AAPL target: $4,000,000/share",
        "label": {
          "id": 1084570,
          "order": 1,
          "color": {
            "red": 0.885322,
            "blue": 0.426327,
            "green": 0.117828,
            "alpha": 1
          },
          "name": "Red"
        }
      }
    }
  ]
}
```



```

    }
  },
  {
    "lastOpenedAt": "2014-01-16T12:39:50+0000",
    "problem": {
      "id": 11135590,
      "title": "API or method to add 3PP",
      "label": {
        "id": 1084571,
        "order": 2,
        "color": {
          "red": 0.094686,
          "blue": 0.834434,
          "green": 0.221608,
          "alpha": 1
        },
        "name": "Blue"
      }
    },
  },
}
},
{
  "lastOpenedAt": "2014-01-16T12:39:50+0000",
  "problem": {
    "id": 12251692,
    "title": "Mechanism to file bugs from iOS without
frequent Apple Connect password prompts",
    "label": {
      "id": 1084572,
      "order": 3,
      "color": {
        "red": 0.063471,
        "blue": 0.210667,
        "green": 0.662984,
        "alpha": 1
      },
      "name": "Green"
    }
  },
}
}
]
}

```

#### 4.22.13 Get Recently Opened Test Suites

##### [A] Description

This API is used to fetch the recent test suites and cases opened in detail view in a Radar Client

**[B] URL Scheme**

GET /test-suites/recent

**[C] Response Attributes**

Key	Description	Data Type
recent	List of recently opened test suites	Array of Test Suites Objects

**Recent Test Suite Objects**

Key	Description	Data Type
testSuite	Test Suites Included in Other Objects	Object
lastOpenedAt	Date and Time when the item was opened by the signed-in user	ISO 8601 Datetime string

**Test Suites Included in Other Objects**

Key	Description	Data Type
id	The test suite ID.	Integer
assignee	Assignee of the test suite. Object will contain dsid, firstName, lastName, email and personType attributes.	Object
component	The problem's component. See 6.5 Components Included in Other Objects.	Object
status	Status of the test suite.	String
title	Title of the test suite.	String
labels	Label details of the test suite. See Response Attributes table in the section 11.1 Get Labels, for description.	Object

**[D] Examples****Get Recent Test Suites**

Client request:

```
GET /test-suites/recent
```

Server response:

HTTP/1.1 200 OK

Status: 200

X-API-Version: 1.6

```
{
  "status":{
    "code":"Success",
    "message":"3 recent test suites returned."
  },
  "recent":[
    {
      "lastOpenedAt":"2014-01-16T12:39:50+0000",
      "testSuite":{
        "id":642907,
        "title":"RWS - Create problem with attachments",
        "label":{
          "id": 1084570,
          "order": 1,
          "color": {
            "red": 0.885322,
            "blue": 0.426327,
            "green": 0.117828,
            "alpha": 1
          },
          "name": "Red"
        }
      }
    },
    {
      "lastOpenedAt":"2014-01-16T12:39:50+0000",
      "testSuite":{
        "id":548062,
        "title":"RWS - Get Enumeration security-list status",
        "label":{
          "id": 1084571,
          "order": 2,
          "color": {
            "red": 0.094686,
            "blue": 0.834434,
```



```

        "green": 0.221608,
        "alpha": 1
      },
      "name": "Blue"
    },
  ],
  {
    "lastOpenedAt": "2014-01-16T12:39:50+0000",
    "testSuite": {
      "id": 548060,
      "title": "RWS - Get Enumeration relation types",
      "label": {
        "id": 1084572,
        "order": 3,
        "color": {
          "red": 0.063471,
          "blue": 0.210667,
          "green": 0.662984,
          "alpha": 1
        },
        "name": "Green"
      }
    }
  }
]
}

```

#### 4.22.14 Get Recently Opened Test Suite Cases

##### [A] Description

This API is used to fetch the recent test suite cases opened in detail view in a Radar Client

##### [B] URL Scheme

GET /test-suites/cases/recent

##### [C] Response Attributes

Key	Description	Data Type
recent	List of recently opened test suite cases	Array of Test Suite Cases Objects



**Recent Test Suite Case Objects**

Key	Description	Data Type
case	Test Suite Case Included in Other Objects	Object
lastOpenedAt	Date and Time when the item was opened by the signed-in user	ISO 8601 Datetime string

**Test Suite Cases Included in Other Objects**

Key	Description	Data Type
id	ID of the Case	Integer
title	title of the Case	String
lastModifiedAt	Last modification date of Case	ISO 8601 Datetime string

**[D] Examples****Get Recent Test Suite Cases**

Client request:

GET /test-suite/cases/recents

Server response:

HTTP/1.1 200 OK

Status: 200

X-API-Version: 1.6

```
{
  "status":{
    "code":"Success",
    "message":"3 recent test suite cases returned."
  },
  "cases":[
    {
      "lastOpenedAt":"2014-01-16T12:39:50+0000",
      "case":{
        "id":1109275,
        "title":"Fetch Enumeration relation type",
        "lastModifiedAt":"2013-12-16T12:39:50+0000"
      }
    }
  ]
}
```



```

    }
  },
  {
    "lastOpenedAt":"2014-01-16T12:39:50+0000",
    "testSuite":{
      "id":1109209,
      "title":"CreateProblem AP",
      "lastModifiedAt":"2013-11-16T12:39:50+000"
    }
  },
  {
    "lastOpenedAt":"2014-01-16T12:39:50+0000",
    "testSuite":{
      "id":1106726,
      "title":"Verify the button title with sprint selectio",
      "lastModifiedAt":"2013-10-16T12:39:50+000"
    }
  }
]
}

```

#### 4.22.15 Get Recently Opened Scheduled Tests

##### [A] Description

This API is used to fetch the recent scheduled tests opened in detail view in a Radar Client

##### [B] URL Scheme

GET /scheduled-tests/recent

##### [C] Response Attributes

Key	Description	Data Type
recent	List of recently opened Scheduled Tests	Array of Scheduled Test Objects

**Recent Scheduled Test Objects**

Key	Description	Data Type
scheduledTest	Scheduled Test Included in Other Objects	Object
lastOpenedAt	Date and Time when the item was ope by the signed-in user	ISO 8601 Datetime string

**Scheduled Tests Included in Other Objects**

Key	Description	Data Type
id	The Scheduled Tests ID.	Integer
assignee	Assignee of the Scheduled Test. Object will contain dsid, firstName, lastName, email and personType attributes.	Object
component	The Scheduled Test's component. See 6.5 Components Included in Other Objects.	Object
status	Status of the Scheduled Test	String
title	Title of the Scheduled Test	String
labels	Label details of the Scheduled Test. See Response Attributes table in the section 11.1 Get Labels, for description.	Object

**[D] Examples****Get Recent Scheduled Tests**

Client request:

```
GET /scheduled-tests/recents
X-Fields-Requested: id,title,label
```



Server response:

HTTP/1.1 200 OK

Status: 200

X-API-Version: 1.6

```
{
  "status":{
    "code":"Success",
    "message":"3 recent scheduled tests returned."
  },
  "recents":[
    {
      "lastOpenedAt":"2014-01-16T12:39:50+0000",
      "scheduledTest":{
        "id":642907,
        "title":"RWS - Create problem with attachments",
        "label":{
          "id": 1084570,
          "order": 1,
          "color": {
            "red": 0.885322,
            "blue": 0.426327,
            "green": 0.117828,
            "alpha": 1
          },
          "name": "Red"
        }
      }
    },
    {
      "lastOpenedAt":"2014-01-16T12:39:50+0000",
      "scheduledTest":{
        "id":548062,
        "title":"RWS - Get Enumeration security-list",
        "label":{
          "id": 1084571,
          "order": 2,
          "color": {
            "red": 0.094686,
            "blue": 0.834434,
            "green": 0.221608,
            "alpha": 1
          },
          "name": "Blue"
        }
      }
    }
  ],
  {
    status,
```



```

    "lastOpenedAt": "2014-01-16T12:39:50+0000",
    "scheduledTest": {
      "id": 548060,
      "title": "RWS - Get Enumeration relation types",
      "label": {
        "id": 1084572,
        "order": 3,
        "color": {
          "red": 0.063471,
          "blue": 0.210667,
          "green": 0.662984,
          "alpha": 1
        },
        "name": "Green"
      }
    }
  ]
}

```

#### 4.22.16 Get Recently Opened Scheduled Test Cases

##### [A] Description

This API is used to fetch the recent scheduled test cases opened in detail view in a Radar Client

##### [B] URL Scheme

GET /scheduled-tests/cases/recent

##### [C] Response Attributes

Key	Description	Data Type
recent	List of recently opened test suites	Array of Scheduled Test Case Objects

#### Recent Scheduled Test Case Objects

Key	Description	Data Type
case	Scheduled Test Case Included in Other Objects	Object



Key	Description	Data Type
lastOpenedAt	Date and Time when the item was ope by the signed-in user	ISO 8601 Datetime string

#### Scheduled Test Cases Included in Other Objects

Key	Description	Data Type
id	ID of the Scheduled Test Case	Integer
title	title of the Scheduled Test Case	String
lastModifiedAt	Last modification date of Scheduled Test Case	ISO 8601 Datetime string

#### [D] Examples

##### Get Recent Scheduled Test Cases

Client request:

```
GET /scheduled-tests/cases/recent
```

Server response:

HTTP/1.1 200 OK

Status: 200

X-API-Version: 1.6

```
{
  "status":{
    "code":"Success",
    "message":"3 recent scheduled test cases returned."
  },
  "cases":[
    {
      "lastOpenedAt":"2014-01-16T12:39:50+0000",
      "case":{
        "id":8251181,
        "title":"Radar needs to behave better when automatically autolaunched",
        "lastModifiedAt":"2013-12-16T12:39:50+0000"
      }
    }
  ],
}
```



```
{
  "lastOpenedAt":"2014-01-16T12:39:50+0000",
  "case":{
    "id":19073137,
    "title":"Start a sprint and execute the resource stats service",
    "lastModifiedAt":"2013-11-16T12:39:50+0000"
  }
},
{
  "lastOpenedAt":"2014-01-16T12:39:50+0000",
  "case":{
    "id":19066660,
    "title":"Verify efforts for ds id when assignee is changed with updates in the efforts",
    "lastModifiedAt":"2013-10-16T12:39:50+0000"
  }
}
]
```

## 4.23 Draft APIs

### 4.23.1 Create Problem Draft

#### [A] Description

Problem Drafts can be created using this API.. This route supports the X-Fields-Requested header for selecting the response object.

#### [B] Schedule

Required for version 1.6

#### [C] URL Scheme

POST /problems/drafts

#### [D] Request Attributes

Key	Description	Data Type	Required
title	Title of the problem. Maximum size of 240 Characters.	String	Y
component	The component that the problem belongs to. (See 6.5 Components Included in Other Objects)	Object	N
description	Description of the problem	String	N



Key	Description	Data Type	Required
diagnosis	Diagnosis of the problem	String	N
classification	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	N
reproducible	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	N
taskOrder	The task order value. Float value can have max of 3 decimal values and 8 digits, but total length should not exceeds 11 characters including radix point.	Float or null	N
configuration	The full configuration information. Maximum size of 1000000 characters.	String or null	N
configuration Summary	The one-line configuration summary. Maximum size of 240 characters.	String or null	N
workaround	A description of the workaround. Maximum size of 1000000 characters.	String or null	N
sourceChanges	Source changes. Maximum size of 1000000 characters.	String or null	N
releaseNotes	The release notes. Maximum size of 1000000 characters.	String or null	N
failedModule	The name of the module that failed. Maximum size of 60 characters.	String or null	N
failureDetail	A one-line summary of the failure. Maximum size of 60 characters.	String or null	N
succinctSummaryRootCause	A detailed summary of the failure. Maximum size of 2000 characters.	String or null	N
actionTaken	A description of the corrective action taken. Maximum size of 2000 characters.	String or null	N
dateNeeded Current	The current value for date needed. (startactualdate).	ISO 8601 date string or null	N
dateNeeded Original	The original, planned value for date needed. (startplanneddate)	ISO 8601 date string or null	N





Key	Description	Data Type	Required
targetCompletionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date string or null	N
targetCompletionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date string or null	N
targetStartDate	The targeted start date. (duedate)	ISO 8601 date string or null	N
isApproved	Has this work been approved? (featureapproved)	Boolean	N
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean	N
isAutoCalculated	Whether to auto-calculate dates and effort from subtasks.	Boolean	N
hasNewAPIImpact	Does this problem involve new API? (featurenewapi)	Boolean	N
hasNewSPIImpact	Feature has new SPI (featurenewspi)	Boolean	N
hasHumanInterfaceImpact	Does the problem have HI impact? (featurehi)	Boolean	N
hasThirdPartyImpact	Feature has third party content (feature3rdparty)	Boolean	N
hasImportExportImpact	Does this problem require import / export review? (featureimpexp)	Boolean	N
hasLocalizationImpact	Does the problem have localization impact? (featureloc)	Boolean	N
hasPatentReviewImpact	Feature needs patent review (featureneedspatent)	Boolean	N
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean	N
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean	N



Key	Description	Data Type	Required
effortCurrentTotalEstimate	Effort, current total estimate, in days. (effortcurtotal)	Float or null	N
effortOriginalTotalEstimate	Effort, original total estimate, in days. (effortinittotal)	Float or null	N
effortPercentComplete	Effort, percent complete, from 0 to 100. (effortpercentcomplete)	Integer or null	N
effortRemaining	Effort, remaining, in days. (effortremain)	Float or null	N
effortExpended	Effort, expended, in days. (effortexpended)	Float or null	N
testCase	Feature test case (featuretextcaseid). Maximum size of 768 characters.	String or null	N
foundInBuild	The name of the component build that this problem was found in. Maximum size of 25 characters.	String or null	N
fixedInBuild	The name of the component build that this problem was fixed in. Maximum size of 25 characters.	String or null	N
verifiedInBuild	The name of the component build that this problem was verified in. Maximum size of 25 characters.	String or null	N
mustBeFixedInBuild	The name of the component build that this problem must be fixed in. Maximum size of 25 characters.	String or null	N
isVerifiedByTester	Was this problem verified by a tester?	Boolean	N
isRegressionRequired	Must this problem be regressed?	Boolean	N
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Maximum size of 2048 characters.	String or null	N

#### [E] Response Attributes

Key	Description	Data Type	Default
draftID	Radar Draft ID	Int	Y
fingerprint	A unique identifier that captures if the problem has changed	String	Y



Key	Description	Data Type	Default
title	Title of the problem. Maximum size of 240 Characters.	String	Y
component	The component that the problem belongs to. (See 6.5 Components Included in Other Objects)	Object or Null	Y
originator	The person who created the problem. (See 7.2 People Included in Other Objects)	Object	Y
description	Description of the Problem Draft	String	Y
diagnosis	Description of the Problem Draft	String	Y
classification	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	Y
reproducible	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	Y
taskOrder	The task order value. Float value can have max of 3 decimal values and 8 digits, but total length should not exceeds 11 characters including radix point.	Float or null	N
configuration	The full configuration information. Maximum size of 1000000 characters.	String or null	N
configuration Summary	The one-line configuration summary. Maximum size of 240 characters.	String or null	N
workaround	A description of the workaround. Maximum size of 1000000 characters.	String or null	N
sourceChanges	Source changes. Maximum size of 1000000 characters.	String or null	N
releaseNotes	The release notes. Maximum size of 1000000 characters.	String or null	N
failedModule	The name of the module that failed. Maximum size of 60 characters.	String or null	N
failureDetail	A one-line summary of the failure. Maximum size of 60 characters.	String or null	N
succinctSummaryRootCause	A detailed summary of the failure. Maximum size of 2000 characters.	String or null	N



Key	Description	Data Type	Default
actionTaken	A description of the corrective action taken. Maximum size of 2000 characters.	String or null	N
dateNeeded Current	The current value for date needed. (startactualdate).	ISO 8601 date string or null	N
dateNeeded Original	The original, planned value for date needed. (startplanneddate)	ISO 8601 date string or null	N
targetCompl etionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date string or null	N
targetCompl etionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date string or null	N
targetStartDa te	The targeted start date. (duedate)	ISO 8601 date string or null	N
isApproved	Has this work been approved? (featureapproved)	Boolean	N
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean	N
isAutoCalcul ated	Whether to auto-calculate dates and effort from subtasks.	Boolean	N
hasNewAPII mpact	Does this problem involve new API? (featurenewapi)	Boolean	N
hasNewSPII mpact	Feature has new SPI (featurenewspi)	Boolean	N
hasHumanIn terfaceImpact	Does the problem have HI impact? (featurehi)	Boolean	N
hasThirdPart yImpact	Feature has third party content (feature3rdparty)	Boolean	N
hasImportEx portImpact	Does this problem require import / export review? (featureimpexp)	Boolean	N
hasLocalizati onImpact	Does the problem have localization impact? (featureloc)	Boolean	N
hasPatentRev iewImpact	Feature needs patent review (featureneedspatent)	Boolean	N



Key	Description	Data Type	Default
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean	N
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean	N
effortCurrentTotalEstimate	Effort, current total estimate, in days. (effortcurtotal)	Float or null	N
effortOriginalTotalEstimate	Effort, original total estimate, in days. (effortinittotal)	Float or null	N
effortPercentComplete	Effort, percent complete, from 0 to 100. (effortpercentcomplete)	Integer or null	N
effortRemaining	Effort, remaining, in days. (effortremain)	Float or null	N
effortExpended	Effort, expended, in days. (effortexpended)	Float or null	N
testCase	Feature test case (featuretextcaseid). Maximum size of 768 characters.	String or null	N
foundInBuild	The name of the component build that this problem was found in. Maximum size of 25 characters.	String or null	N
fixedInBuild	The name of the component build that this problem was fixed in. Maximum size of 25 characters.	String or null	N
verifiedInBuild	The name of the component build that this problem was verified in. Maximum size of 25 characters.	String or null	N
mustBeFixedInBuild	The name of the component build that this problem must be fixed in. Maximum size of 25 characters.	String or null	N
isVerifiedByTester	Was this problem verified by a tester?	Boolean	N
isRegressionRequired	Must this problem be regressed?	Boolean	N
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Maximum size of 2048 characters.	String or null	N



Key	Description	Data Type	Default
createdAt	Date and time the Problem Draft was created	ISO 8601 date string or null	N
lastModified At	Date and time the Problem Draft was last modified	ISO 8601 date string or null	N

#### [F] Example

Client request:

```
POST /problems/1.6/drafts
X-Fields-Requested: title, draftID
Content-Type: application/json
{
  "title": "Can't delete problems"
}
```

Server response:

```
HTTP/1.1 201 Created
Content-Type: application/json
{
  "status": {
    "code": "Success",
    "message": "1 Problem Draft created."
  },
  "draft": {
    "title": "Can't delete problems",
    "draftID": 1001
  }
}
```

### 4.23.2 Delete Problem Draft

#### [A] Description

Deletes Problem Draft from Radar if the signed-in user is the creator.

#### [B] Schedule

Required for version 1.6

#### [C] URL Scheme

```
DELETE /problems/drafts/<draft-id>
```

**[D] Example**

Client request:

```
DELETE /problems/drafts/1001
```

Server response:

```
HTTP/1.1 200 Success
```

**4.23.3 Get All Drafts****[A] Description**

Returns sign-in user's open Problem Draft objects.

**[B] Schedule**

Required for version 1.6

**[C] URL Scheme**

```
GET /problems/drafts
```

**[C] Response Attributes**

Key	Description	Data Type	Default
draftID	Radar Draft ID	Int	Y
fingerprint	A unique identifier that captures if the problem has changed	String	Y
title	Title of the problem. Maximum size of 240 Characters.	String	Y
component	The component that the problem belongs to. (See 6.5 Components Included in Other Objects)	Object	Y
createdAt	Date and time the Problem Draft was created	ISO 8601 date string or null	Y
lastModified At	Date and time the Problem Draft was last modified	ISO 8601 date string or null	Y

**[D] Example**



Client request:

```
GET /problems/drafts
```

Server response:

```
HTTP/1.1 200 OK
```

```
{
  "status":{
    "code":"Success",
    "message":"1 Problem Draft returned."
  },
  "drafts":[
    {
      "draftID":1001,
      "title":"Can't delete a problem",
      "fingerprint":"787c96n2",
      "component": {
        "name": "Radar",
        "version": "Automation"
      }
      "createdAt":"2014-01-16T12:39:50+0000",
      "lastModifiedAt":"2014-01-16T12:39:50+0000",
    }
  ]
}
```

#### 4.23.4 Get Problem Draft by ID

##### [A] Description

Returns a single Problem Draft object identified by it ID.

##### [B] Schedule

Required for version 1.6

##### [C] URL Scheme

```
GET /problems/drafts/<draft-ID>
```

##### [D] Response Attributes

Key	Description	Data Type	Default
draftID	Radar Draft ID	Int	Y





Key	Description	Data Type	Default
fingerprint	A unique identifier that captures if the problem has changed	String	Y
title	Title of the problem. Maximum size of 240 Characters.	String	Y
component	The component that the problem belongs to. (See 6.5 Components Included in Other Objects)	Object or Null	Y
originator	The person who created the problem. (See 7.2 People Included in Other Objects)	Object	Y
description	Description of the Problem Draft	String	Y
diagnosis	Description of the Problem Draft	String	Y
classification	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	Y
reproducible	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	Y
taskOrder	The task order value. Float value can have max of 3 decimal values and 8 digits, but total length should not exceeds 11 characters including radix point.	Float or null	N
configuration	The full configuration information. Maximum size of 1000000 characters.	String or null	N
configuration Summary	The one-line configuration summary. Maximum size of 240 characters.	String or null	N
workaround	A description of the workaround. Maximum size of 1000000 characters.	String or null	N
sourceChanges	Source changes. Maximum size of 1000000 characters.	String or null	N
releaseNotes	The release notes. Maximum size of 1000000 characters.	String or null	N
failedModule	The name of the module that failed. Maximum size of 60 characters.	String or null	N
failureDetail	A one-line summary of the failure. Maximum size of 60 characters.	String or null	N



Key	Description	Data Type	Default
succinctSummaryRootCause	A detailed summary of the failure. Maximum size of 2000 characters.	String or null	N
actionTaken	A description of the corrective action taken. Maximum size of 2000 characters.	String or null	N
dateNeededCurrent	The current value for date needed. (startactualdate).	ISO 8601 date string or null	N
dateNeededOriginal	The original, planned value for date needed. (startplanneddate)	ISO 8601 date string or null	N
targetCompletionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date string or null	N
targetCompletionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date string or null	N
targetStartDate	The targeted start date. (duedate)	ISO 8601 date string or null	N
isApproved	Has this work been approved? (featureapproved)	Boolean	N
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean	N
isAutoCalculated	Whether to auto-calculate dates and effort from subtasks.	Boolean	N
hasNewAPIImpact	Does this problem involve new API? (featurenewapi)	Boolean	N
hasNewSPIImpact	Feature has new SPI (featurenewspi)	Boolean	N
hasHumanInterfaceImpact	Does the problem have HI impact? (featurehi)	Boolean	N
hasThirdPartyImpact	Feature has third party content (feature3rdparty)	Boolean	N
hasImportExportImpact	Does this problem require import / export review? (featureimpexp)	Boolean	N



Key	Description	Data Type	Default
hasLocalizationImpact	Does the problem have localization impact? (featureloc)	Boolean	N
hasPatentReviewImpact	Feature needs patent review (featureneedspatent)	Boolean	N
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean	N
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean	N
effortCurrentTotalEstimate	Effort, current total estimate, in days. (effortcurtotal)	Float or null	N
effortOriginalTotalEstimate	Effort, original total estimate, in days. (effortinittotal)	Float or null	N
effortPercentComplete	Effort, percent complete, from 0 to 100. (effortpercentcomplete)	Integer or null	N
effortRemaining	Effort, remaining, in days. (effortremain)	Float or null	N
effortExpended	Effort, expended, in days. (effortexpended)	Float or null	N
testCase	Feature test case (featuretextcaseid). Maximum size of 768 characters.	String or null	N
foundInBuild	The name of the component build that this problem was found in. Maximum size of 25 characters.	String or null	N
fixedInBuild	The name of the component build that this problem was fixed in. Maximum size of 25 characters.	String or null	N
verifiedInBuild	The name of the component build that this problem was verified in. Maximum size of 25 characters.	String or null	N
mustBeFixedInBuild	The name of the component build that this problem must be fixed in. Maximum size of 25 characters.	String or null	N
isVerifiedByTester	Was this problem verified by a tester?	Boolean	N
isRegressionRequired	Must this problem be regressed?	Boolean	N



Key	Description	Data Type	Default
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Maximum size of 2048 characters.	String or null	N
createdAt	Date and time the Problem Draft was created	ISO 8601 date string or null	N
lastModified At	Date and time the Problem Draft was last modified	ISO 8601 date string or null	N

### [E] Example

Client request:

```
GET /problems/drafts/1001
X-Fields-Requested: id,title
```

Server response:

```
HTTP/1.1 200 OK
{
  "status":{
    "code":"Success",
    "message":"1 Problem Draft returned."
  },
  "draft":{
    "id":1001,
    "title":"Can't delete a problem"
  }
}
```

## 4.23.5 Update Problem Draft

### [A] Description

Updates any properties on Problem Drafts with the values supplied in the request objects.

### [B] Schedule

Required for version 1.6

### [C] URL Scheme

```
POST /problems/drafts/<draft-id>
```

### [D] Request Attributes



Key	Description	Data Type	Required
title	Title of the problem. Maximum size of 240 Characters.	String	Y
component	The component that the problem belongs to. (See 6.5 Components Included in Other Objects)	Object	N
description	Description of the problem	String	N
diagnosis	Diagnosis of the problem	String	N
classification	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	N
reproducible	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	N
taskOrder	The task order value. Float value can have max of 3 decimal values and 8 digits, but total length should not exceeds 11 characters including radix point.	Float or null	N
configuration	The full configuration information. Maximum size of 1000000 characters.	String or null	N
configuration Summary	The one-line configuration summary. Maximum size of 240 characters.	String or null	N
workaround	A description of the workaround. Maximum size of 1000000 characters.	String or null	N
sourceChanges	Source changes. Maximum size of 1000000 characters.	String or null	N
releaseNotes	The release notes. Maximum size of 1000000 characters.	String or null	N
failedModule	The name of the module that failed. Maximum size of 60 characters.	String or null	N
failureDetail	A one-line summary of the failure. Maximum size of 60 characters.	String or null	N
succinctSummaryRootCause	A detailed summary of the failure. Maximum size of 2000 characters.	String or null	N
actionTaken	A description of the corrective action taken. Maximum size of 2000 characters.	String or null	N



Key	Description	Data Type	Required
dateNeeded Current	The current value for date needed. (startactualdate).	ISO 8601 date string or null	N
dateNeeded Original	The original, planned value for date needed. (startplanneddate)	ISO 8601 date string or null	N
targetCompl etionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date string or null	N
targetCompl etionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date string or null	N
targetStartDa te	The targeted start date. (duedate)	ISO 8601 date string or null	N
isApproved	Has this work been approved? (featureapproved)	Boolean	N
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean	N
isAutoCalcul ated	Whether to auto-calculate dates and effort from subtasks.	Boolean	N
hasNewAPII mpact	Does this problem involve new API? (featurenewapi)	Boolean	N
hasNewSPII mpact	Feature has new SPI (featurenewsapi)	Boolean	N
hasHumanIn terfaceImpact	Does the problem have HI impact? (featurehi)	Boolean	N
hasThirdPart yImpact	Feature has third party content (feature3rdparty)	Boolean	N
hasImportEx portImpact	Does this problem require import / export review? (featureimpexp)	Boolean	N
hasLocalizati onImpact	Does the problem have localization impact? (featureloc)	Boolean	N
hasPatentRev iewImpact	Feature needs patent review (featureneedspatent)	Boolean	N



Key	Description	Data Type	Required
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean	N
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean	N
effortCurrentTotalEstimate	Effort, current total estimate, in days. (effortcurtotal)	Float or null	N
effortOriginalTotalEstimate	Effort, original total estimate, in days. (effortinittotal)	Float or null	N
effortPercentComplete	Effort, percent complete, from 0 to 100. (effortpercentcomplete)	Integer or null	N
effortRemaining	Effort, remaining, in days. (effortremain)	Float or null	N
effortExpended	Effort, expended, in days. (effortexpended)	Float or null	N
testCase	Feature test case (featuretextcaseid). Maximum size of 768 characters.	String or null	N
foundInBuild	The name of the component build that this problem was found in. Maximum size of 25 characters.	String or null	N
fixedInBuild	The name of the component build that this problem was fixed in. Maximum size of 25 characters.	String or null	N
verifiedInBuild	The name of the component build that this problem was verified in. Maximum size of 25 characters.	String or null	N
mustBeFixedInBuild	The name of the component build that this problem must be fixed in. Maximum size of 25 characters.	String or null	N
isVerifiedByTester	Was this problem verified by a tester?	Boolean	N
isRegressionRequired	Must this problem be regressed?	Boolean	N
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Maximum size of 2048 characters.	String or null	N

#### [E] Response Attributes



Key	Description	Data Type	Default
draftID	Radar Draft ID	Int	Y
fingerprint	A unique identifier that captures if the problem has changed	String	Y
title	Title of the problem. Maximum size of 240 Characters.	String	Y
component	The component that the problem belongs to. (See 6.5 Components Included in Other Objects)	Object or Null	Y
originator	The person who created the problem. (See 7.2 People Included in Other Objects)	Object	Y
description	Description of the Problem Draft	String	Y
diagnosis	Description of the Problem Draft	String	Y
classification	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	Y
reproducible	An enumerated string value. Possible values can be fetched using 10.1 Get Field Enumeration.	String	Y
taskOrder	The task order value. Float value can have max of 3 decimal values and 8 digits, but total length should not exceeds 11 characters including radix point.	Float or null	N
configuration	The full configuration information. Maximum size of 1000000 characters.	String or null	N
configuration Summary	The one-line configuration summary. Maximum size of 240 characters.	String or null	N
workaround	A description of the workaround. Maximum size of 1000000 characters.	String or null	N
sourceChanges	Source changes. Maximum size of 1000000 characters.	String or null	N
releaseNotes	The release notes. Maximum size of 1000000 characters.	String or null	N
failedModule	The name of the module that failed. Maximum size of 60 characters.	String or null	N
failureDetail	A one-line summary of the failure. Maximum size of 60 characters.	String or null	N





Key	Description	Data Type	Default
succinctSummaryRootCause	A detailed summary of the failure. Maximum size of 2000 characters.	String or null	N
actionTaken	A description of the corrective action taken. Maximum size of 2000 characters.	String or null	N
dateNeededCurrent	The current value for date needed. (startactualdate).	ISO 8601 date string or null	N
dateNeededOriginal	The original, planned value for date needed. (startplanneddate)	ISO 8601 date string or null	N
targetCompletionCurrent	The current value for target completion date. (targetcurcompdate)	ISO 8601 date string or null	N
targetCompletionOriginal	The original, planned value for target completion date. (targetorigcompdate)	ISO 8601 date string or null	N
targetStartDate	The targeted start date. (duedate)	ISO 8601 date string or null	N
isApproved	Has this work been approved? (featureapproved)	Boolean	N
isUmbrella	Is this an umbrella problem? (featureumbrella)	Boolean	N
isAutoCalculated	Whether to auto-calculate dates and effort from subtasks.	Boolean	N
hasNewAPIImpact	Does this problem involve new API? (featurenewapi)	Boolean	N
hasNewSPIImpact	Feature has new SPI (featurenewspi)	Boolean	N
hasHumanInterfaceImpact	Does the problem have HI impact? (featurehi)	Boolean	N
hasThirdPartyImpact	Feature has third party content (feature3rdparty)	Boolean	N
hasImportExportImpact	Does this problem require import / export review? (featureimpexp)	Boolean	N



Key	Description	Data Type	Default
hasLocalizationImpact	Does the problem have localization impact? (featureloc)	Boolean	N
hasPatentReviewImpact	Feature needs patent review (featureneedspatent)	Boolean	N
hasConfidentialContentImpact	Does this problem have confidential content? (featureconfidential)	Boolean	N
hasOpenSourceImpact	Does the problem have open source code? (featurehasopensource)	Boolean	N
effortCurrentTotalEstimate	Effort, current total estimate, in days. (effortcurtotal)	Float or null	N
effortOriginalTotalEstimate	Effort, original total estimate, in days. (effortinittotal)	Float or null	N
effortPercentComplete	Effort, percent complete, from 0 to 100. (effortpercentcomplete)	Integer or null	N
effortRemaining	Effort, remaining, in days. (effortremain)	Float or null	N
effortExpended	Effort, expended, in days. (effortexpended)	Float or null	N
testCase	Feature test case (featuretextcaseid). Maximum size of 768 characters.	String or null	N
foundInBuild	The name of the component build that this problem was found in. Maximum size of 25 characters.	String or null	N
fixedInBuild	The name of the component build that this problem was fixed in. Maximum size of 25 characters.	String or null	N
verifiedInBuild	The name of the component build that this problem was verified in. Maximum size of 25 characters.	String or null	N
mustBeFixedInBuild	The name of the component build that this problem must be fixed in. Maximum size of 25 characters.	String or null	N
isVerifiedByTester	Was this problem verified by a tester?	Boolean	N
isRegressionRequired	Must this problem be regressed?	Boolean	N



Key	Description	Data Type	Default
buildInfo	The serial number or build info. (Equivalent to partNumber in the DB). Maximum size of 2048 characters.	String or null	N
createdAt	Date and time the Problem Draft was created	ISO 8601 date string or null	N
lastModified At	Date and time the Problem Draft was last modified	ISO 8601 date string or null	N

### [F] Example

Client request:

```
POST /problems/drafts/1001
X-Fields-Requested: title, draftID
Content-Type: application/json
{
  "title": "Can't delete problems in Mail"
}
```

Server response:

```
HTTP/1.1 200 Created
Content-Type: application/json
{
  "status":{
    "code":"Success",
    "message":"1 Problem Draft updated."
  },
  "draft": {
    "title": "Can't delete problems in Mail",
    "draftID": 1001
  }
}
```

### 4.23.6 Get Drafts Usage

#### [A] Description

Describes the server-side Draft allocation and how many draft problems the user has created.

#### [B] Schedule

Required for version 1.6

**[C] URL Scheme**

GET /problems/drafts

**[D] Response Attributes**

Key	Description	Data Type	Default
draftsCount	Quantity of Radar Problem Drafts saved by the user	Integer	Y
allocationCount	Quantity of Radar Problem Drafts allocated according to Radar's server-side rules	Integer	Y

**[D] Example**

Client request:

GET /problems/drafts/usage

Server response:

HTTP/1.1 200 OK

```
{
  "status":{
    "code":"Success",
    "message":"Problem Draft usage data returned."
  },
  "usage":{
    "draftsCount":1,
    "allocationCount":250
  }
}
```



## 5. KEYWORDS

Keywords in radar may be referenced by name or id when associating them with a problem. The API in this section provides a method for looking up keywords by name.

### 5.1 Get Keyword by ID

#### [A] Description

This API provides a method for looking up one or more keywords by ID. Since keywords with the same name can exist in different contexts, this will ensure the correct keyword is returned.

#### [C] URL Scheme

```
GET /keywords/<keyword_id>[,<keyword_id>]
```

#### [D] Request Attributes

There is no request body: the ID or IDs are specified in the URL. A list of fields can be requested in the `X-Fields-Requested` header.

#### [E] Response Attributes

The response is an object or an array of objects. By default, it includes just the id and name attributes. This table describes all the attributes that can be requested.

Key	Description	Data Type
id	The unique ID of the keyword.	Integer
name	The name of the keyword	String
type	Type of keyword. One of: "Normal", "Personal", "Shared", "Global", "Restricted", "Restricted Remove".	String
description	The long description of the keyword.	String
source	One of "Public", "Personal", or the name of the shared keyword set that the keyword comes from.	String
isCopiedOnClone	Indicates whether this keyword will be copied when the problem is cloned.	Boolean
isClosed	Indicates whether the keyword is closed.	Boolean
component	The keyword's component, if public. (See <a href="#">6.1 Find Components for description</a> ).	Object



## [F] Examples

### Find a keyword by ID:

Client request:

```
GET /keywords/48910
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: application/json; charset=utf-8
{
  "id": 48910,
  "name": "Radar6.1 IQA Both"
}
```

## 5.2 Find Keywords

### [A] Description

This API provides a method to search for keywords.

The default number of rows returned is 2000. You may increase the number of rows returned by using the “X-rowlimit” header. Note that a large rowlimit value with too broadly defined query criteria can lead to a connection timeout. If this happens, focus your criteria more narrowly.

### [C] URL Scheme

```
POST /keywords/find
```

### [D] Request Attributes

Key	Description	Data Type
name	A keyword name prefix. Maximum Length is 30 characters.	String
id	Id of the keyword	Integer
type	Type of keyword. One of: “Normal”, “Personal”, “Shared”, “Global”, “Restricted”, “Restricted Remove”. If not present, will search all keyword types (just like the GET form).	String



Key	Description	Data Type
component	The component that contains this keyword. See <a href="#">6.5 Components Included in Other Objects</a> . This key only applies to “type” values of null, “Normal”, “Global”, “Restricted”, or “Restricted Remove”.	Object
fieldsRequested	An array of fields that the response objects should contain.	Array
isClosed	Is the keyword closed?	Boolean

### [E] Response Attributes

The response is an array of keyword objects. By default, it includes just the id and name attributes. This table describes all the attributes that can be requested.

Key	Description	Data Type
id	The unique ID of the keyword.	Integer
name	The name of the keyword. Maximum length is 30 Characters	String
type	Type of keyword. One of: “Normal”, “Personal”, “Shared”, “Global”, “Restricted”, “Restricted Remove”.	String
description	The long description of the keyword. Maximum length is 80 Characters	String
source	One of “Public”, “Personal”, or the name of the shared keyword set that the keyword comes from.	String
isCopiedOnClone	Indicates whether this keyword will be copied when the problem is cloned.	Boolean
isClosed	Indicates whether the keyword is closed.	Boolean
component	The keyword’s component, if public. (See <a href="#">6.5 Components Included in Other Objects</a> ).	Object

### [F] Examples

#### Keyword search by name prefix:

Client request:

```
POST /keywords/find
X-API-Version: 1.0
```



```
Content-Type: application/json;charset=UTF-8
{
  "name": "Radar"
}
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "id": 48910,
    "name": "Radar6.1 IQA Both"
  },
  {
    "id": 48911,
    "name": "Radar6.1 IQA CLI"
  }
]
```

## 5.3 Keywords Included in Other Objects

### [A] Description

Except where specified otherwise, a keyword included in another object (such as [4.5.1 Get Problem Keywords List](#)) will include the attributes specified here.

### [B] Schedule

Required for version 1.0

### [C] Attributes

Key	Description	Data Type
name	The keyword's name.	String
id	The keyword's ID (for disambiguation).	Integer





## 6. COMPONENTS

This section covers APIs related to managing components.

### 6.1 Get Component by Name and Version

#### [A] Description

This API provides a method to retrieve one or more components by component name and/or version.

#### [B] URL Scheme

```
GET /components/<component_name>
GET /components/<component_name>/<component_version>
```

#### [C] Request Attributes

The request has no body, but fields other than default can be selected with the X-Fields-Requested as in [3.1 Get Problem By ID](#).

#### [D] Response Attributes

The response is an object which can contain the following Key. The ones marked as default will be included by default, whereas others are available through the X-Fields-Requested header.

Key	Description	Data Type	Default
id	ID of the component	Integer	Y
name	name of the component. Maximum length is 30 characters.	String	Y
version	version of the component. Maximum length is 30 characters.	String	Y
description	Description of the component. Maximum length is 240 characters.	String	Y
isClosed	Is component closed?	Boolean	Y
isRestricted	Is component restricted?	Boolean	Y
isRootLevel	Is component root level?	Boolean	N
isOpenToExternals	Does the component allow externals?	Boolean	N
isDropBox	Is component a drop box?	Boolean	N



Key	Description	Data Type	Default
doesInheritAccess Groups	Does the component inherit access groups?	Boolean	N
doesInheritBuilds AndMilestones	Does the component inherit builds and milestones?	Boolean	N
doesInheritDescriptionTemplates	Does the component inherit description templates?	Boolean	N
parent	The component's parent component. parent object will contain all the fields mentioned in these table except followOnComponent and subcomponents.	Object	N
subcomponents	The component's subcomponents. parent object will contain all the fields mentioned in these table except followOnComponent and parent.	Array of objects	N
owner	Owner of the component. See <a href="#">7.2 People Included in Other Objects</a> .	Object	N
epm	EPM for the component. See <a href="#">7.2 People Included in Other Objects</a> .	Object	N
screener	Screener of the component. See <a href="#">7.2 People Included in Other Objects</a> .	Object	N
integrator	Integrator of the component. See <a href="#">7.2 People Included in Other Objects</a> .	Object	N
builder	Builder of the component. See <a href="#">7.2 People Included in Other Objects</a> .	Object	N
verifier	Verifier of the component. See <a href="#">7.2 People Included in Other Objects</a> .	Object	N
treeLimited	Does the component problem is tree specific	Boolean	N
followOnComponent	Follow-on component of component. Follow-on component object will contain all the fields mentioned in these table except parent and subcomponents.	Object	N

### [E] Examples

#### Get Component by name and version:

Client request:



```
GET /components/Radar/6.15
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: application/json; charset=utf-8
```

```
{
  "id": 20496,
  "name": "Radar",
  "version": "6.15",
  "description": "Enhancements",
  "isClosed": true,
  "isProduct": true,
  "isRestricted": false
}
```

#### Get Component with subcomponents:

Client request:

```
GET /components/Radar/6.15
X-API-Version: 1.0
X-Fields-Requested: id,name,version,subcomponents
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: application/json; charset=utf-8
```

```
{
  "id": 20496,
  "name": "Radar",
  "version": "6.15",
  "subcomponents": [
    {
      "id": 429103,
      "name": "Radar",
      "version": "Dev-6.15",
      "subcomponents": []
    }
  ]
}
```



## 6.2 Find Components

### [A] Description

This API provides a method to search for components.

The default number of rows returned is 2000. You may increase the number of rows returned by using the “x-rowlimit” header. Note that a large rowlimit value with too broadly defined query criteria can lead to a connection timeout. If this happens, focus your criteria more narrowly.

### [C] URL Scheme

POST /components/find

### [D] Request Attributes

Key	Description	Data Type
id	ID of the component	Integer
name	Name of the component. Maximum length is 30 characters.	String
version	Version of the component. Maximum length is 30 characters.	String
description	Description of the component. Maximum length is 240 characters.	String
includeSubs	Include sub component in search	Boolean
isRootLevel	Include root level component in search	Boolean
isRestricted	Include restricted component in search	Boolean
isDropBox	Include drop box in search	Boolean
doesInheritAccessGroups	Search on whether the component inherits access groups.	Boolean
doesInheritBuildsAndMilestones	Search on whether the component inherits builds and milestones.	Boolean
doesInheritDescriptionTemplates	Search on whether the component inherits description templates.	Boolean
isOpenToExternals	Search on whether the component allows externals.	Boolean
hasExternals	Search on whether the component currently has externals.	Boolean
isClosed	Search for only closed or open components.	Boolean



Key	Description	Data Type
build	Searches builds attached to the component. Please refer below Build / Milestone / Keyword Object details.	Object
milestone	Searches milestones attached to the component. Please refer below Build / Milestone / Keyword Object details.	Object
event	Searches event attached to the component. Please refer below Build / Milestone / Keyword Object details.	Object
keyword	Searches keywords attached to the component. Please refer below Build / Milestone / Keyword Object details.	Object
workGroup	Searches the work groups associated with the component.	String
accessGroup	Searches the access groups associated with the component.	String
sharedLabel	Searches the shared labels associated with the component.	String
sharedKeyword	Searches the shared keywords associated with the component, specified as a string or id.	String or Integer
personnel	Include person DSID and type with operator. Refer Personnel Object for more details.	Object
followOnComponent	The follow-on Component. (See <a href="#">6.5 Components Included in Other Objects</a> )	Object
fieldsRequested	A list of fields that should be included in the component objects. This will override the default fields. See <a href="#">6.1 Get Component By Name and Version</a> .	Array of strings

#### Build/Milestone/Event/Keyword Object

Key	Description	Data Type
name	name of the build / milestone / event / keyword. Maximum length is 30 characters.	String
isClosed	Is build / milestone / event / keyword closed?	Boolean

#### Personnel Object

Key	Description	Data Type
dsid	DSID of the person	Integer
operator	Operator for comparison. It includes any one of (gt, lt, eq, ne)	String
personRole	Role of the person in component	String



## [D] Response Attributes

The response contains an array of component objects with the same Key as [6.1 Get Component By Name and Version](#).

## [E] Examples

**Find unrestricted components with a given build and EPM:**

Client request:

```
POST /components/find
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "build": {
    "isClosed": false,
    "name": "testMilestone"
  },
  "isRestricted": false,
  "personnel": {
    "dsid": 8794,
    "operator": "eq",
    "personRole": "epm"
  }
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "description": "component for radar 6.15 version",
    "id": 20496,
    "isClosed": false,
    "isRestricted": false,
    "name": "Radar",
    "version": "6.15"
  },
  {
    "description": "component for radarWeb 6.15 version",
    "id": 20106,
    "isClosed": false,
    "isRestricted": true,
    "name": "RadarWeb",
    "version": "6.15"
  }
]
```

**Find Radar Automation component tree:**

Client request:

```
POST /components/find
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "fieldsRequested": [
    "name",
    "version",
    "subcomponents"
  ],
  "name": "Radar",
  "version": "Automation"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "name": "Radar",
    "subcomponents": [
      {
        "name": "Radar",
        "subcomponents": [],
        "version": "Dev-Automation"
      }
    ],
    "version": "Automation"
  }
]
```

**Find all components involving Alan Ewalt:**

Client request:

```
POST /components/find
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "personnel": {
    "dsid": 8794,
    "operator": "eq",
    "personRole": "all"
  }
}
```

Server response:



```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "name": "Alaska",
    "version": "0.0",
    ...
  },
  ...
]
```

**Find Component with fieldsRequested attribute:**

Client request:

```
POST /components/find
X-API-Version: 1.4
{
  "name": "Radar",
  "version": "7.0",
  "fieldsRequested":["id", "name", "version", "description"]
}
```

Server response:

```
HTTP/1.1 200 Ok
Status: 200
X-API-Version: 1.4
[
  {
    "id": 23423,
    "name": "Radar",
    "version": "7.0",
    "description": "Component for all Radar 7.0 bugs"
  }
]
```

**Find Component with fieldsRequested attribute and X-Fields-Requested header:**

Client request:

```
POST /components/find
X-API-Version: 1.4
X-Fields-Requested: id,name,version,isRootLevel
{
  "name": "Radar",
  "version": "7.0",
  "fieldsRequested":["id", "name", "version", "description"]
}
```

Server response:





```
HTTP/1.1 400 Bad Request
Status: 400
X-API-Version: 1.4
{
  "message": "Mutually exclusive fields X-Fields-Requested and
fieldsRequested are passed in request.",
  "title": "Mutually exclusive fields",
  "status": "400 Bad Request",
  "help": "View documentation at http://radar.apple.com/"
}
```

## 6.3 Component Builds, Milestones and Event

This set of APIs deals with adding and finding builds ,milestones or event for a component.

### 6.3.1 Add Build or Event to Component

#### [A] Description

This API provides a method to add a build or event to a component.

#### [B] URL Scheme

POST /components/<name>/<version>/<builds\_or\_events>

#### [C] Request Attributes

The URL includes the component name and version, as well as the object type, which can be either “builds” or “events”.

Key	Description	Data Type
name	name of the item. Maximum length is 25 characters.	String
beginsAt	begin date of the item	ISO 8601 date string
endsAt	end date of the item	ISO 8601 date string

#### [D] Response Attributes

The response attributes match those of the objects in 6.3.2 Get Component Builds or Milestones.

#### [E] Examples

**Add Component Build:**



Client request:

```
POST /components/Radar/Automation/builds
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "beginsAt": "2011-12-21T11:39:00-0800",
  "endsAt": "2011-01-22T00:30:00-0800",
  "name": "RadarBuild"
}
```

Server response:

```
HTTP/1.1 201 Created
X-API-Version: 1.0
Status: 201
Content-Type: application/json;charset=UTF-8
{
  "beginsAt": "2011-12-21T11:39:00-0800",
  "endsAt": "2011-01-22T00:30:00-0800",
  "id": 8097342,
  "isClosed": false,
  "isInherited": false,
  "name": "RadarBuild"
}
```

#### Add Component Event:

Client request:

```
POST /components/Radar/Automation/events
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "beginsAt": "2011-12-21T11:39:00-0800",
  "endsAt": "2011-01-22T00:30:00-0800",
  "name": "RadarEvent"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "beginsAt": "2011-12-21T18:39:00+0000",
  "endsAt": "2011-01-22T08:30:00+0000",
  "id": 8097342,
  "isClosed": false,
  "isInherited": false,
  "name": "RadarEvent"
}
```



### 6.3.2 Get Component Builds or Events or Milestone

#### [A] Description

This API provides a method to fetch all builds or milestone or events attached to a component.

For milestone two more attributes “groupName” and “isRestricted” is added in 1.1 version of Web Services. It will contain comma separated name of all the Access Group added to milestone and Boolean value for isRestricted attribute. If no access group is added then “groupName” attribute will contain a null value.

#### [B] URL Scheme

GET /components/<name>/<version>/<builds\_or\_milestones\_or\_events>

#### [C] Request Attributes

The request has no body. The URL includes the component name and version and the object type, which can be either “builds”, “milestones” or “events”.

#### [D] Response Attributes

Key	Description	Data Type
id	ID of the item	Integer
name	Name of the item. Maximum length is 25 characters.	String
beginsAt	Begin date of the item	ISO 8601 date string
endsAt	End date of the item	ISO 8601 date string
isClosed	Whether the item is closed or not (only applies to milestones)	Boolean
isInherited	Whether this item has been inherited from an ancestor component.	Boolean
isRestricted	Whether the component is restricted. Fetched only for milestone.	Boolean
groupName	String containing group name separated by comma. Fetched only for restricted milestone.	String

#### [E] Examples

##### Get Component Builds:



Client request:

```
GET /components/Radar/Automation/builds
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "beginsAt": "2009-02-17T02:31:00-0800",
    "endsAt": "2009-02-17T02:31:00-0800",
    "id": 123,
    "isClosed": true,
    "isInherited": true,
    "name": "6.0"
  },
  {
    "beginsAt": "2009-02-17T02:31:00-0800",
    "endsAt": "2009-02-17T02:31:00-0800",
    "id": 124,
    "isClosed": true,
    "isInherited": true,
    "name": "6.0.1"
  },
  {
    "beginsAt": "2009-02-17T02:31:00-0800",
    "endsAt": "2009-02-17T02:31:00-0800",
    "id": 125,
    "isClosed": true,
    "isInherited": true,
    "name": "6.0.2"
  },
  ...
]
```

### Get Component Events:

Client request:

```
GET /components/Radar/Automation/events
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
```



```

    "beginsAt": "2009-02-17T02:31:00+0000",
    "endsAt": "2009-02-17T02:31:00+0000",
    "id": 123,
    "isClosed": true,
    "isInherited": true,
    "name": "6.0a"
  },
  ...
]

```

### 6.3.3 Add Milestone to Component

#### [A] Description

This API is used to create a new restricted milestone for a component. If the request contains attribute `isRestricted` as `true` then `accessGroup` and `privileges` attribute must be passed in request. If `"isRestricted"` attribute is passed as `false` then normal milestone is created without any restriction.

#### [B] Schedule

Required for version 1.1

#### [C] URL Scheme

POST `components/<component-name>/<component-version>/milestones`

#### [D] Request Attributes

Key	Description	Data Type
name	Name of the milestone. Maximum length is 25 characters.	String
beginsAt	Begin date of the milestone	ISO8601 Date String
endsAt	End date of the milestone	ISO8601 Date String
isRestricted	Is milestone restricted	Boolean
accessGroup	Name of the access group	String
privileges	Privileges for the access group. It can have any one of Read Only, Assignable, Reassignable(Tree-limited), Reassignable(Full).	String

#### [E] Examples

##### Add Milestone to Component:



Client request:

```
POST /components/Radar/Automation/milestones
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "accessGroup": "Radar7.1",
  "beginsAt": "2011-12-21T11:39:00-0800",
  "endsAt": "2011-01-22T00:30:00-0800",
  "isRestricted": true,
  "name": "RadarEvent",
  "privileges": "Read Only"
}
```

Server response:

```
HTTP/1.1 201 Created
Date: Wed, 04 Jun 2014 09:12:45 GMT
Content-Length: 176
Content-Type: application/json;charset=UTF-8
{
  "id": 186247,
  "groupName": null,
  "beginsAt": "2011-01-21T19:39:00+0000",
  "isRestricted": false,
  "endsAt": "2011-01-22T08:30:00+0000",
  "name": "test",
  "isInherited": false,
  "isClosed": false
}
```

### 6.3.4 Add Access Group to Milestone

#### [A] Description

This API is used to add a access group to milestone of a component. Upon adding an Access Group to a milestone which is not restricted will make it to a restricted milestone after adding a group.

#### [B] Schedule

Required for version 1.1

#### [C] URL Scheme

```
POST components/<component-name>/<component-version>/milestones/access-
groups
```

**[D] Request Attributes**

Key	Description	Data Type
milestone	Name of milestone. Maximum length is 25 characters.	String
accessGroup	Name of the access group	String
privileges	Privileges for the access group. It can have any one of Read Only, Assignable, Reassignable(Tree-limited), Reassignable(Full).	String

**[E] Examples****Add Access Group to Milestone of Component:**

Client request:

```
POST /components/Radar/Automation/milestones/access-groups

X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "milestone": "RadarMilestone",
  "accessGroup": "Radar7.1",
  "privileges": "Read Only"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

**6.3.5 Remove Access Group From Milestone****[A] Description**

This API is used to remove a access group from milestone of a component. If restricted milestone have only one Access Group added then it cannot be removed, If tried to remove then an error message will be shown.

**[B] Schedule**

Required for version 1.1

**[C] URL Scheme**

```
DELETE components/<component-name>/<component-version>/milestones/  
<milestone-name>/access-groups/<access-group-name>[<privileges>]
```

**[D] Examples****Remove Access Group from Milestone of Component:**

Client request:

```
DELETE /components/Radar/Automation/milestones/RadarMilestone/access-  
groups/Radar7.1/Assignable  
Content-Type: application/json; charset=UTF-8  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content  
Status: 204  
X-API-Version: 1.0
```

**6.3.6 Edit Component Event, Milestone , or Build****[A] Description**

This API is used to edit any existing component event, milestone, or build data. All fields mentioned in Request parameters are optional, but request body should contain any one field to update, otherwise an error message will be shown. Modifying events, milestones, or builds on a parent component is unsupported.

If the logged-in user does not have access to given component then HTTP 403 Forbidden status will be shown as error.

Validations are performed to make sure the existing event, milestone, or build exists and that the new event, milestone, or build name does not already exist. If new name is already in use by some other event, milestone, or build then an error message will be shown.

Upon successful update, a HTTP 200 Ok status will be returned without any response body.

**[B] Schedule**

Required for version 1.5

**[C] URL Scheme**

```
PUT /components/<name>/<version>/<builds | milestones | events>/<name>
```





## [D] Request Parameters

All below attributes are optional, but at least one should be passed in request body.

Key	Description	Data Type	Required
name	New name of event. Maximum length is 32 char	String	N
beginsAt	Event begin date	ISO Date time String	N
endsAt	Event end date	ISO Date time String	N
isClosed	Flag to show event is closed or not.	Boolean	N

## [E] Examples

### Edit existing component event:

Client request:

```
PUT /components/Radar/Automation/events/1.4
Content-Type: application/json; charset=UTF-8
{
  "name": "1.4.1",
  "beginsAt": "2014-02-28T12:00:00",
  "endsAt": "2014-03-07T12:00:00"
}
```

Server response:

```
HTTP/1.1 200 Ok
Status: 200
```

### Error for forbidden component access:

Client request:

```
PUT /components/Radar/Automation/events/1.4
Content-Type: application/json; charset=UTF-8
{
  "name": "1.4.1",
  "beginsAt": "2014-02-28T12:00:00",
  "endsAt": "2014-03-07T12:00:00"
}
```

Server response:

```
HTTP/1.1 403 Forbidden
Status: 403
```



```
Content-Type: application/json; charset=utf-8
{
  "status": "403 Forbidden",
  "title": "Component Access Forbidden",
  "message": "You do not have access to the component with name 'Radar'
and version 'Automation'.",
  "help": "View documentation at http://radar.apple.com"
}
```

**Error for Duplicate Event Name:**

Client request:

```
PUT /components/Radar/Automation/events/1.4
Content-Type: application/json; charset=UTF-8
{
  "name": "1.4.1",
  "beginsAt": "2014-02-28T12:00:00",
  "endsAt": "2014-03-07T12:00:00"
}
```

Server response:

```
HTTP/1.1 409 Conflict
Status: 409
Content-Type: application/json; charset=utf-8
{
  "status": "409 Conflict",
  "title": "Event Name Conflict",
  "message": "The event '1.4.1' is already defined in this component or
being inherited from its parent. Please choose another event name.",
  "help": "View documentation at http://radar.apple.com"
}
```

**Error for empty request:**

Client request:

```
PUT /components/Radar/Automation/events/1.4
Content-Type: application/json; charset=UTF-8
{}
```

Server response:

```
HTTP/1.1 400 Bad Request
Status: 409
Content-Type: application/json; charset=utf-8
{
  "status": "400 Bad Request",
  "title": "Missing Request Parameters",
  "message": "The request is missing at least one request parameter.",
  "help": "View documentation at http://radar.apple.com"
}
```



### Error for beginsAt greater then endsAt date value:

Client request:

```
PUT /components/Radar/Automation/events/1.4
Content-Type: application/json; charset=UTF-8
{
  "name": "1.4.1",
  "beginsAt": "2014-02-28T12:00:00",
  "endsAt": "2014-02-27T12:00:00"
}
```

Server response:

```
HTTP/1.1 409 Conflict
Status: 409
Content-Type: application/json; charset=utf-8
{
  "status": "409 Conflict",
  "title": "Date Conflict",
  "message": "Begin date should not be greater than the End Date.",
  "help": "View documentation at http://radar.apple.com"
}
```

## 6.3.7 Remove Component Event, Milestone , or Build

### [A] Description

This API is used to remove any existing component event, milestone, or build. Inherited events, milestones, or builds from parent component cannot be remove using this API.

If logged-in user does not have access to given component then HTTP 403 Forbidden status will be shown as error.

Upon successful removal, a HTTP 200 OK status will be returned without any response body.

### [B] Schedule

Required for version 1.5

### [C] URL Scheme

```
DELETE /components/<name>/<version>/<builds | milestones | events>/
<name>
```

### [D] Examples

#### Remove existing component event:

Client request:



```
DELETE /components/Radar/Automation/events/1.4
```

Server response:

```
HTTP/1.1 200 Ok
Status: 200
```

**Error for forbidden component access:**

Client request:

```
DELETE /components/Radar/Automation/events/1.4
```

Server response:

```
HTTP/1.1 403 Forbidden
Status: 403
Content-Type: application/json; charset=utf-8
{
  "status": "403 Forbidden",
  "title": "Component Access Forbidden",
  "message": "You do not have access to the component with name 'Radar'
and version 'Automation'.",
  "help": "View documentation at http://radar.apple.com"
}
```

**Error for given event not found:**

Client request:

```
DELETE /components/Radar/Automation/events/1.4
```

Server response:

```
HTTP/1.1 404 Not Found
Status: 404
Content-Type: application/json; charset=utf-8
{
  "status": "404 Not Found",
  "title": "Event Not Found",
  "message": "Event with name '1.4' not found for component name 'Radar'
and version 'Automation'.",
  "help": "View documentation at http://radar.apple.com"
}
```

### 6.3.8 Component Builds and Milestones Included in Other Objects

#### [A] Description

Except where specified otherwise, a build or milestone included in another object (such as the assignee in [3.1 Get Problem By ID](#)) will include the attributes specified here.

**[B] Schedule**

Required for version 1.0

**[C] Attributes**

Key	Description	Data Type
name	The item name.	String
component	The item's containing component. (See <a href="#">6.5 Components Included in Other Objects</a> )	Object

## 6.4 Component Bundles

This Set of APIs provides a method to manage component bundles. The user can pass either bundle name or ID in the URL parameter. If the datatype of the received parameter is number then it will be considered as ID else as name. If there are duplicate bundle names, appropriate error message will be send.

### 6.4.1 Create Component Bundle

**[A] Description**

This API provides a method to create a component bundle. The bundle owner will be the currently logged in user. On success, the service responds with the created bundle as described in [6.4.4 Get Component Bundle](#).

**[B] Schedule**

Required for version 1.0

**[C] URL Scheme**

POST /component-bundles

**[D] Request Parameters**

Parameter	Description	Data type
name	Name of the bundle. Maximum length is 80 characters	String
description	Description of the bundle. Maximum length is 80 characters	String
isActive	Is bundle active.	Boolean



Parameter	Description	Data type
isPublic	Is bundle public.	Boolean

## [E] Examples

### Create Component Bundle:

Client request:

```
POST /component-bundles
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "description": "Component Bundle for Radar automation",
  "isActive": true,
  "isPublic": false,
  "name": "Radar Web"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
{
  "description": "Component Bundle for Radar automation",
  "id": 1656,
  "isActive": true,
  "isPublic": false,
  "name": "Radar Web",
  "owner": {
    "dsid": 1234,
    "firstName": "Joel",
    "lastName": "Young"
  }
}
```

## 6.4.2 Add Component To Bundle

### [A] Description

This API provides a method to add component to a existing component bundle. The request body must contain name and version of component which need to be added to component bundle. 'includeSubcomponents' is a optional attribute in request body, if not included in request then it will be taken as false by default.

**[B] Schedule**

Required for version 1.0

**[C] URL Scheme**

POST /component-bundles/<bundle-name or bundle-id>/components

**[D] Request Parameters**

Parameter	Description	Data Type
name	Name of component. Maximum length is 30 characters	String
version	Version of component. Maximum length is 30 characters	String
includeSubcomponents	Are subcomponents to be included?	Boolean

**[E] Examples****Add Component to a Bundle:**

Client request:

```
POST /component-bundles/1656/components
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
{
  "name": "Radar",
  "version": "6.15"
  "includeSubcomponents": true
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
Content-Type: application/json;charset=UTF-8
X-API-Version: 1.0
```

**6.4.3 Remove Component from Bundle****[A] Description**

This API provides a method to remove component from component bundle.

**[B] Schedule**

Required for version 1.0

**[C] URL Scheme**

```
DELETE /component-bundles/<bundle-name or bundle-id>/components/  
<component-name>/<component-version>
```

**[D] Examples****Remove Component From a Bundle:**

Client request:

```
DELETE /component-bundles/1656/components/Radar/6.15  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content  
Status: 204  
X-API-Version: 1.0
```

**6.4.4 Get Component Bundle****[A] Description**

This API provides a method to get the component bundle details with attached components to it. The request URL contains the bundle ID or name. If request contains bundle ID then response will contain a json object containing bundle details and if request contains bundle name then it will return array of object containing all the bundle satisfying name passed in request. User need to pass header “X-Component-Children” as true for fetching all children components of the the components.

**[B] Schedule**

Required for version 1.0

**[C] URL Scheme**

```
GET /component-bundles/<bundle-name or bundle-id>
```

**[D] Response Parameters**

Object	Description	Data Type
id	ID of the bundle requested	Integer





Object	Description	Data Type
name	Name of the bundle. Maximum length is 80 characters	String
description	Description of the bundle. Maximum length is 80 characters	String
components	Components added to the bundle. (See ComponentObject for component bundle table below for description)	Array of objects
owner	Person object of the owner (Described in ???)	Object
isActive	Is bundle active.	Boolean
isPublic	Is bundle public.	Boolean

#### Component Object for Component bundle

Object	Description	Data Type
id	ID of the component	Integer
name	Name of the component. Maximum length is 30 characters	String
version	Version name of the component. Maximum length is 30 characters	String
isComponentTreeIncluded	Boolean that describes whether the entire component tree is included in the bundle	boolean
subcomponents	Components children(See Component children Object below)	Array of objects

#### ComponentChildrenObject for Component bundle

Object	Description	Data type
id	ID of the component	Integer
name	Name of the component	String
version	Version name of the component	String



Object	Description	Data type
subcomponents	Components children(See Component children Object)	Array of objects

## [E] Examples

### Get Component Bundle with Bundle Name:

Client request:  
 GET /component-bundles/test%20Bundle  
 X-API-Version: 1.2

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "isActive": true,
    "id": 2174,
    "description": "descc",
    "name": "test bundle",
    "owner": {
      "lastName": "Tester1",
      "email": "radartester01@gmail.com",
      "type": "Employee",
      "firstName": "Radar",
      "dsid": 1118580968
    },
    "components": [
      {
        "id": 80440,
        "isComponentTreeIncluded": false,
        "name": "Test",
        "version": "1"
      }
    ],
    "isPublic": false
  },
  {
    "isActive": true,
    "id": 2185,
    "description": "desc for test bundle x2",
    "name": "test bundle x2",
    "owner": {
      "lastName": "Tester1",
      "email": "radartester01@gmail.com",
      "type": "Employee",
      "firstName": "Radar",
      "dsid": 1118580968
    }
  },
]
```



```

    "components": [
      {
        "id": 509925,
        "isComponentTreeIncluded": false,
        "name": "test component 123",
        "version": "123"
      },
      {
        "id": 509924,
        "isComponentTreeIncluded": false,
        "name": "test component xyz",
        "version": "xyz"
      }
    ],
    "isPublic": true
  }
]

```

### Get Component Bundle with Bundle ID:

Client request:

```

GET /component-bundles/1401
X-API-Version: 1.3

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8

```

```

{
  "isActive": true,
  "id": 1401,
  "description": "A Bundle for rounding up Audio responsibilities",
  "name": "Audio Bundle",
  "owner": {
    "lastName": "Carlinsky",
    "email": "ecarlinsky@apple.com",
    "type": "Employee",
    "firstName": "Eric",
    "dsid": 89859139
  },
  "components": [
    {
      "id": 141251,
      "isComponentTreeIncluded": true,
      "name": "AudioHW",
      "version": "New Bugs"
    },
    {
      "id": 220627,
      "isComponentTreeIncluded": true,
      "name": "CPU Software Yellow",
      "version": "Sound New"
    }
  ]
}

```



```

    {
      "id": 370684,
      "isComponentTreeIncluded": true,
      "name": "Mac Acoustic Design",
      "version": "New"
    }
  ],
  "isPublic": true
}

```

### Get Component Bundle with X-Component-Children:

Client request:

```

GET /component-bundles/1401
X-API-Version: 1.3
X-Component-Children: true

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
{
  "isActive": true,
  "id": 1401,
  "description": "A Bundle for rounding up Audio responsibilities",
  "name": "Audio Bundle",
  "owner": {
    "lastName": "Carlinsky",
    "email": "ecarlinsky@apple.com",
    "type": "Employee",
    "firstName": "Eric",
    "dsid": 89859139
  },
  "components": [
    {
      "id": 141251,
      "isComponentTreeIncluded": true,
      "subcomponents": [
        {
          "id": 483592,
          "subcomponents": [],
          "name": "AudioHW",
          "version": "Analog Devices"
        },
        {
          "id": 248660,
          "subcomponents": [],
          "name": "AudioHW",
          "version": "Characterizations"
        }
      ]
    }
  ]
}

```



```
"id": 483591,
"subcomponents": [],
"name": "AudioHW",
"version": "Cirrus"
},
{
  "id": 338723,
  "subcomponents": [],
  "name": "AudioHW",
  "version": "Documentation"
},
{
  "id": 474184,
  "subcomponents": [],
  "name": "AudioHW",
  "version": "Investigations"
},
{
  "id": 343594,
  "subcomponents": [],
  "name": "AudioHW",
  "version": "Limits Tables"
},
{
  "id": 476755,
  "subcomponents": [],
  "name": "AudioHW",
  "version": "Maxim"
},
{
  "id": 483593,
  "subcomponents": [],
  "name": "AudioHW",
  "version": "NXP"
},
{
  "id": 483590,
  "subcomponents": [],
  "name": "AudioHW",
  "version": "Texas Instruments"
},
{
  "id": 291623,
  "subcomponents": [],
  "name": "AudioHW",
  "version": "Wolfson"
},
{
  "id": 338703,
  "subcomponents": [],
  "name": "Microphone",
  "version": "Acoustic Test"
},
{
  "id": 338644,
```



```

        "subcomponents": [],
        "name": "Speaker",
        "version": "Acoustic Test"
    }
],
"name": "AudioHW",
"version": "New Bugs"
},
{
    "id": 220627,
    "isComponentTreeIncluded": true,
    "subcomponents": [
        {
            "id": 228829,
            "subcomponents": [],
            "name": "CPU Software Yellow",
            "version": "Sound Dupe"
        },
        {
            "id": 311545,
            "subcomponents": [],
            "name": "CPU Software Yellow",
            "version": "Sound Ethernet"
        },
        {
            "id": 341223,
            "subcomponents": [],
            "name": "CPU Software Yellow",
            "version": "Sound Firmware"
        },
        {
            "id": 253906,
            "subcomponents": [],
            "name": "CPU Software Yellow",
            "version": "Sound Orig"
        },
        {
            "id": 431944,
            "subcomponents": [],
            "name": "CPU Software Yellow",
            "version": "Sound Random"
        },
        {
            "id": 460105,
            "subcomponents": [],
            "name": "CPU Software Yellow.....",
            "version": "close this"
        },
        {
            "id": 388964,
            "subcomponents": [],
            "name": "DGB",
            "version": "Cross functional"
        }
    ]
},

```



```

    "name": "CPU Software Yellow",
    "version": "Sound New"
  },
  {
    "id": 370684,
    "isComponentTreeIncluded": true,
    "subcomponents": [
      {
        "id": 474185,
        "subcomponents": [],
        "name": "Mac Acoustic Design",
        "version": "Documentation"
      },
      {
        "id": 344124,
        "subcomponents": [],
        "name": "Mac Acoustic Design",
        "version": "Investigations"
      },
      {
        "id": 472374,
        "subcomponents": [],
        "name": "Mac Acoustic Design",
        "version": "Microphone Module"
      },
      {
        "id": 472373,
        "subcomponents": [],
        "name": "Mac Acoustic Design",
        "version": "Speaker Module"
      }
    ],
    "name": "Mac Acoustic Design",
    "version": "New"
  }
],
"isPublic": true
}

```

#### Error message for Bundle ID which does not exists in DB

Client request:

```
GET /component-bundles/213123
```

Server response:

```

HTTP/1.1 404 Not Found
Content-Type: application/json; charset=utf-8
{
  "message": "Component Bundle with ID '213123' does not found in
Database.",
  "title": "Component bundle not found",
  "help": "View documentation at http://radar.apple.com/",

```



```
"status": "404 Not Found"
}
```

### 6.4.5 Modify Component Bundle

#### [A] Description

This API provides a method to update the details of component bundle.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
POST /component-bundles/<bundle-name or bundle-id>
```

#### [D] Request Parameters

Parameter	Description	Data Type
name	Name of the bundle. If this is changed, it will also change the url path.. Maximum length is 80 characters	String
description	New Description of the bundle. Maximum length is 80 characters	String
isActive	Whether the bundle is active or not	Boolean
isPublic	Whether the bundle is public or not	Boolean

#### [E] Examples

##### Modify a Component Bundle:

Client request:

```
POST /component-bundles/1656
Content-Type: application/json;charset=UTF-8
X-API-Version: 1.0
{
  "description": "Component Bundle for radar",
  "isActive": true,
  "isPublic": true,
  "name": "Radar automation"
}
```

Server response:





HTTP/1.1 201 Created  
Status: 201  
X-API-Version: 1.0

## 6.4.6 Get All Component Bundle List

### [A] Description

This API is used to get all accessible component bundle list to logged-in user. This API will return only basic details of all component bundle which are mentioned in Response parameters. Response will be array of all component bundle accessible to user.

A response with an empty array is sent to users with no access to any component bundle.

### [B] Schedule

Required for version 1.5

### [C] URL Scheme

GET /component-bundles

### [D] Response Parameters

Response will be array of object containing below attributes.

Key	Description	Data Type
id	Id of component bundle	Integer
name	Name of Component Bundle	String
description	Description of component bundle	String
owner	Owner of component bundle.	Person Object
isActive	Flag to show bundle is active or not.	Boolean
isPublic	Flag to show bundle is public or not.	Boolean

### [E] Examples

#### Get all component bundles:

Client request:

GET /component-bundles



Server response:

```
HTTP/1.1 200 Ok
Status: 200
Content-Type: application/json; charset=UTF-8
[
  {
    "id": 1353,
    "name": "AppleTV",
    "description": "AppleTV",
    "owner": {
      "lastName": "Young",
      "firstName": "James",
      "dsid": 1118580968,
      "email": "xxx@apple.com",
      "type": "Employee"
    },
    "isActive": true,
    "isPublic": true
  }
  ..... more objects
]
```

## 6.5 Fetch component tree

### [A] Description

The API provides the method to fetch entire tree for a given component. User can pass multiple components in the request. Closed components and respective subcomponents will not be displayed when the 'includeClosed' flag is set as false. Root component details will be displayed irrespective of the 'includeClosed' flag value. If the user passed invalid component details, then error message will be added. If the component don't have any subcomponents then subcomponents attribute will be empty array.

### [B] Schedule

Required for version 1.3

### [C] URL Scheme

POST /components/component-tree

### [D] Request Attributes

Key	Description	Data Type
components	Component to fetch entire tree. (See <a href="#">ComponentObject for ComponentTree Request</a> table below for description)	Array of Objects



Key	Description	Data Type
includeClosed	Is include Closed Components?	Boolean

#### ComponentObject for ComponentTree Request:

Key	Description	Data Type
name	Name of the Component	String
version	Version of the Component	String

#### [E] Response Attributes

Response is an array of Component Objects. Each Component Object will contains the following attributes.

Key	Description	Data Type
id	ID of the Component	Integer
name	Name of the Component	String
version	Version of the Component	String
description	Description of the Component	String
isRestricted	Is component restricted?	Boolean
isClosed	Is component closed?	Boolean
subcomponents	Subcomponents of the Component. (The attributes are same as the Response Component Object attributes)	Array of Objects

#### [F] Examples

##### (i) Fetching Entire Component Tree:

Client Request:

```
POST /components/component-tree
X-API-Version: 1.3
Content-Type: application/json; charset=utf-8
{
  "components":
  [
    {
      "name": "Radar",
      "version": "WebServices v1.2"
```



```

    },
    {
      "name": "iWork",
      "version": "Casper"
    }
  ],
  "includeClosed": true
}

```

Server response:

HTTP/1.1 200 OK

Status: 200

Content-Type: application/json; charset=utf-8

```

[
  {
    "id": 509984,
    "isRestricted": false,
    "description": "Component for Radar WS",
    "name": "Radar",
    "subcomponents": [
      {
        "id": 509986,
        "isRestricted": false,
        "description": "Component for WS-1.2",
        "name": "Radar",
        "subcomponents": [
          {
            "id": 509988,
            "isRestricted": false,
            "description": "Component for WS-1.2 -Developement",
            "name": "Radar",
            "subcomponents": [],
            "isClosed": false,
            "version": "WebServices v1.2 - Dev"
          },
          {
            "id": 509990,
            "isRestricted": false,
            "description": "Component for WS-1.2 - iQA",
            "name": "Radar",
            "subcomponents": [],
            "isClosed": true,
            "version": "WebServices v1.2 - iQA"
          }
        ],
        "isClosed": false,
        "version": "WebServices v1.2"
      },
      {

```



```

        "id": 509985,
        "isRestricted": false,
        "description": "Component for WS-1.3",
        "name": "Radar",
        "subcomponents": [
            {
                "id": 509987,
                "isRestricted": false,
                "description": "Component for WS-1.3 -Developement",
                "name": "Radar",
                "subcomponents": [],
                "isClosed": true,
                "version": "WebServices v1.3 - Dev"
            }
        ],
        "isClosed": false,
        "version": "WebServices v1.3"
    },
    {
        "id": 391463,
        "isRestricted": true,
        "description": "Released as 1.3 (1D49 is GM)",
        "name": "iWork",
        "subcomponents": [],
        "isClosed": true,
        "version": "Casper"
    }
]

```

**(ii) Invalid Component detail passed in the Request Body:**

Client Request:

```

POST /components/component-tree
X-API-Version: 1.3
Content-Type: application/json; charset=utf-8
{
    "components":
    [
        {
            "name": "Radar",
            "version": "WebServices v1"
        },
        {
            "name": "iWork",
            "version": "Casper"
        }
    ]
}

```



```

    }
    ],
    "includeClosed": false
}

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "error": "No component found with this name and version",
    "name": "Radar",
    "version": "WebServices v1"
  },
  {
    "id": 391463,
    "isRestricted": true,
    "description": "Released as 1.3 (1D49 is GM)",
    "name": "iWork",
    "subcomponents": [],
    "isClosed": true,
    "version": "Casper"
  }
]

```

### (iii)Root Component is Closed:

When 'includeClosed' flag is set as false, the Root component and respective open children details will be displayed even if the root component is closed.

Client Request:

```

POST /components/component-tree
X-API-Version: 1.3
Content-Type: application/json; charset=utf-8
{
  "components":
  [
    {
      "name": "Radar",
      "version": "WebServices v1.2"
    },
    {
      "name": "iWork",
      "version": "Casper"
    }
  ],
  "includeClosed": false
}

```

Server response:



```

HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "id": 509984,
    "isRestricted": false,
    "description": "Component for Radar WS",
    "name": "Radar",
    "subcomponents":
      [
        {
          "id": 509986,
          "isRestricted": false,
          "description": "Component for WS-1.2",
          "name": "Radar",
          "subcomponents":
            [
              {
                "id": 509988,
                "isRestricted": false,
                "description": "Component for WS-1.2 Developement",
                "name": "Radar",
                "subcomponents": [],
                "isClosed": false,
                "version": "WebServices v1.2 - Dev"
              }
            ],
          "isClosed": false,
          "version": "WebServices v1.2"
        }
      ],
    "isClosed": true,
    "version": "WebServices"
  },
  {
    "id": 391463,
    "isRestricted": true,
    "description": "Released as 1.3 (1D49 is GM)",
    "name": "iWork",
    "subcomponents": [],
    "isClosed": true,
    "version": "Casper"
  }
]

```

## 6.6 Fetch component root details

### [A] Description

This API fetches the Root-level Component Owner information by a given Radar ID.



## [B] Schedule

Required for version 1.0

## [C] URL Scheme

GET /components/component-root/<problemID>

## [D] Response Attributes

Key	Description	Data Type
id	ID of the component	Integer
name	The component's name	String
version	The component's version.	String
description	The component's description	String
isRestricted	Component is restricted?	Boolean
isClosed	Component is closed?	Boolean
owner	Person object of the owner	Object

## [E] Examples

Client Request:

```
GET /components/component-root/12312498
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
```

```
{
  "id": 90641,
  "isRestricted": false,
  "description": "Top level component for all Radar components
(versins 3.0.1 and later) (componentID = 90641)",
  "name": "Radar (new bugs)",
  "owner": {
    "lastName": "Ewalt",
    "email": "ewalt@apple.com",
    "type": "Employee",
    "firstName": "Alan",
    "dsid": 8794
  },
},
```





```
"isClosed": false,  
"version": "All"  
}
```

## 6.7 Components Included in Other Objects

### [A] Description

Except where specified otherwise, a component included in another object (such as [3.1 Get Problem By ID](#)) will include the attributes specified here.

### [B] Schedule

Required for version 1.0

### [C] Attributes

Key	Description	Data Type
name	The component's name.	String
version	The component's version.	String



## 7. PEOPLE

This section covers API related to finding a person.

### 7.1 Find People

#### [A] Description

This API provides a method to find a person based on any request parameter provided. The response of this API contains the array of person object satisfying request parameters.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

POST /people/find

#### [D] Request Parameters

The Request body will contain any of the below parameters. The `personType` and `company` fields cannot be the only search criteria in the request body.

Parameter	Description	Data Type
dsid	Directory services ID of the person	Integer
firstName	First name of the person	String
lastName	Last name of the person	String
type	Person type can take one of "Internal", "External" and "All"	String
company	The name of the company the person is associated with.	String
department	The name of the department the person is associated with.	String
phone	Phone number of the person	String
email	The email address of the person to be found.	String
includeInactive	Include inactive people or not.	Boolean

#### [E] Response Parameters

Parameter	Description	Data Type
-----------	-------------	-----------



dsid	Directory services ID of the person	Integer
firstName	First name of the person	String
lastName	Last name of the person	String
type	Person type can take one of "Internal", "External" and "All"	String
company	The name of the company the person is associated with.	String
department	The name of the department the person is associated with.	String
phone	Phone number of the person	String
email	The email address of the person to be found.	String
isActive	Include inactive people or not.	Boolean

## [F] Examples

### Find Person:

The Response will contain an array of all parameters specified above as request parameter except active for includeInactive parameter.

Client request:

```
POST /people/find
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "firstName": "radar",
  "includeInactive": false,
  "personType": "Internal"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "company": null,
    "department": "R&D IS&T",
    "dsid": "2002006588",
    "email": null,
    "firstName": "Radar",
    "isActive": true,
    "lastName": "TestArex01",
    "type": "Independent Contractor",
```



```

    "phone": null
  },
  {
    "company": null,
    "department": "R&D IS&T",
    "dsid": "2002021017",
    "email": null,
    "firstName": "radar",
    "isActive": true,
    "lastName": "satodev04",
    "type": "Independent Contractor",
    "phone": null
  },
  {
    "company": null,
    "department": "R&D IS&T",
    "dsid": "2002002220",
    "email": null,
    "firstName": "radar",
    "isActive": true,
    "lastName": "SharedUAT03",
    "type": "Independent Contractor",
    "phone": null
  }
]

```

## 7.2 People Included in Other Objects

### [A] Description

Except where specified otherwise, a person included in another object (such as the assignee in [3.1 Get Problem By ID](#)) will include the attributes specified here.

### [B] Schedule

Required for version 1.0

### [C] Attributes

Key	Description	Data Type
dsid	The directory services ID.	Integer
firstName	The person's first name.	String
lastName	The person's last name.	String
email	The person's primary email.	String
type	The type of person record, such as "Employee".	String



## 7.3 Fetching details of logged in person

### [A] Description

This API will give the details of the logged in person. This api will not take any input and the details of the person will be returned identified by the DSID from the authentication mode i.e ds cookie, Opaque token or the Spnego auth mode.

On a successful response, the HTTP status is 200. When an error occurs, an appropriate HTTP response code is used, along with a JSON representation of the error.

### [B] Schedule

Required for version 1.3

### [C] URL Scheme

```
GET /people/find/current-user
```

### [D] Response Parameter

Parameter	Description	Data type
dsid	Directory services ID of the person	Integer
firstName	First name of the person	String
lastName	Last name of the person	String
type	Type of person	String
email	The email address of the person to be found.	String

### [E] Examples

Client request:

```
GET /people/find/current-user
X-API-Version: 1.3
```

Server response:

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Expires: ???
Content-Length: XXXX
Status: 200
Content-Type: application/json; charset=utf-8
```

```
{
  "dsid": 2002780164,
  "firstName": "Radar23",
```



```
"lastName": "Test",  
"type": "Independent Contractor",  
"email": "radar23\_test@apple.com"  
}
```



## 8. TEST SUITES AND SCHEDULED TESTS

This Section covers APIs related to Test Suite and Scheduled test.

### 8.1 TestSuite

This Section covers APIs related to test suite.

#### 8.1.1 Create TestSuite

##### [A] Description

This API provides a method to create a new test suite. Response will contain default attributes of GetTestSuite data API after successful creation of test suite.

##### [B] Schedule

Required for version 1.0

##### [C] URL Scheme

POST /test-suites

##### [D] Request Parameters

Parameter	Description	Data Type
title	Title of the test suite. Maximum length is 240 characters	String
component	A Component Object	Component Object
geography	Region of the test suite.	String
priority	Priority of the test suite.	Integer
trackName	Track name of the test suite.	String
applicationName	Application name to which test suite belongs. . Maximum length is 30 characters	String
complexity	Complexity of the test suite.	String
author	DSID of author of the test suite.	Integer
status	Status of the test suite.	String
diagnosis	Diagnosis of the test suite.	String



Parameter	Description	Data Type
category	Category of the test suite.. Maximum length is 30 characters	String
changeID	ChangeID of the test suite.. Maximum length is 100 characters	String
masterData	Master data of the test suite.. Maximum length is 1000000 characters	String
prerequisites	Prerequisites of the test suite.. Maximum length is 4000 characters	String
isActive	Active status of the test suite.	Boolean
isRegression	Regression status of the test suite.	Boolean

## [E] Examples

### Create TestSuite:

Client request:

```
POST /test-suites
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "priority": 4,
  "author": 1118580968,
  "status": "In Progress",
  "geography": "ALAC",
  "trackName": "Admin",
  "complexity": "low",
  "diagnosis": "Diagnosis Text",
  "title": "Radar WebServices sample test suite",
  "component": { "name": "Radar (new bugs)",
  "version": "All" },
  "applicationName": "radar",
  "category": "Admin",
  "changeID": "Radar",
  "masterData": "Master Data Text",
  "prerequisites": "Prerequisites Data Text",
  "isActive": true,
  "isRegression": true
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```





```
Content-Type: application/json; charset=utf-8
{
  "status": "In Progress",
  "complexity": "Low",
  "applicationName": "radar",
  "isRegression": true,
  "trackName": "Admin",
  "changeID": "Radar",
  "suiteID": 430536,
  "lastModifiedAt": "2013-02-19T14:39:00+0530",
  "title": "Radar WebServices sample test suite",
  "category": "Admin",
  "geography": "ALAC",
  "component": {
    "name": "Radar (new bugs)",
    "version": "All"
  },
  "priority": 4,
  "createdAt": "2013-02-19T14:39:00+0530",
  "isActive": true
}
```

### 8.1.2 Add Case to TestSuite

#### [A] Description

This API provides a method to add new case to test suite.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

POST /test-suites/cases

#### [D] Request Parameters

Parameter	Description	Data Type
suiteID	ID of the test suite.	Integer
title	Case Title for the new test suite case. Maximum length is 240 characters	String
caseNumber	Case Number for the new test suite case.	Integer
data	Case Data for the new test case. Maximum length is 1000000 characters	String



Parameter	Description	Data Type
instructions	Case Instruction for the new test case. Maximum length is 4000 characters	String
expectedResult	Expected result for the new test case. Maximum length is 4000 characters	String
expectedTime	Expected time for the new test case.	String
summary	Case Summary for the new test case. Maximum length is 4000 characters	String
priority	Case priority for the new test case.	Integer

**[E] Response Parameters**

Parameter	Description	Data Type
caseID	ID of the test suite case which was created.	Integer

**[F] Examples****Add Case to Test Suite:**

Client request:

```
POST /test-suites/cases
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "data": "data",
  "instructions": "instructions",
  "caseNumber": 1,
  "priority": 4,
  "summary": "case summary",
  "title": "test case 1",
  "expectedResult": "expected result",
  "expectedTime": "00:59:59",
  "suiteID": 239635
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
{
  "caseID": 9749169
}
```



### 8.1.3 Set TestSuite Data

#### [A] Description

This API provides methods to set attributes in test suite.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

PUT /test-suites/<suite-ID>

#### [D] Request Parameters

Parameter	Description	Data Type
title	Title of the test suite. Maximum length is 240 characters	String
component	A Component Object	Component Object
status	Status of the test suite.	String
priority	Priority of the test suite.	Integer
geography	Region of the test suite.	String
trackName	Track name of the test suite.	String
applicationName	Application name to which test suite belongs. Maximum length is 30 characters	String
complexity	Complexity of the test suite.	String
category	Category of the test suite. Maximum length is 30 characters	String
changeID	Change ID of the test suite. Maximum length is 100 characters	String
isActive	Test Suite is active or not.	Boolean
isRegression	Test Suite regression is set or not.	Boolean



Parameter	Description	Data Type
author	DSID of author of the test suite.	Integer
assignee	DSID of assignee of the test suite.	Integer
diagnosis	Diagnosis of the test suite.	String
masterData	Master Data of the test suite. Maximum length is 1000000 characters	String
prerequisites	Prerequisites of the test suite. Maximum length is 4000 characters	String
keywords	Array of keyword name or keyword ID	Array of String / Integer
labelID	LabelID of the label. To remove label from test suite 'null' should be used.	Integer or null

## [E] Examples

### Set Test Suite Data:

Request body will contain only those parameter which has to be set others are excluded from the body.

Client request:

```
PUT /test-suites/239635
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "geography": "Australia",
  "status": "Approved",
  "applicationName": "radar"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

## 8.1.4 Set TestSuite Case Data

### [A] Description

This API provides methods to set details of test suite case.

**[B] Schedule**

Required for version 1.0

**[C] URL Scheme**

PUT /test-suites/<suite-ID>/cases/<case-number>

**[D] Request Parameters**

Parameter	Description	Data Type
title	Title of the test suite case. Maximum length is 240 characters	String
data	Case Data for case. Maximum length is 1000000 characters	String
instructions	Instructions for the case. Maximum length is 4000 characters	String
expectedResult	Expected result of the case. Maximum length is 4000 characters	String
expectedTime	Expected time to complete case testing.	String
summary	Summary of the case. Maximum length is 4000 characters	String
priority	Priority of the case.	Integer
component	A Component Object	Component Object
author	DSID of author of the test suite case.	Integer
keywords	Array of keyword name or keyword ID	Array of String/ Integer
relatedProblems	Array of related problem object	Array of Related Problem
security	Array of security object	Array of security object

**Related Problem Object**

Parameter	Description	Data Type
id	ProblemID which need to be related to case	Integer
relationType	Relation type of related problem	String



## Security Object

Parameter	Description	Data Type
type	Security type which need to added. Type can be either of Person, Work-Group or Access-Group.	String
name	Name of Group. This field should be included if type is either of Work-Group or Access-Group.	String
dsid	DSID of person. This field should be included if type is Person.	Integer

### [E] Examples

#### Set Test Suite Case Data:

Request body will contain only those parameter which has to be set others are excluded from the body.

Client request:

```
PUT /test-suites/239635/cases/1
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "data": "radar web service",
  "title": "test case for radar automation"
}
```

#### Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### Set Test Suite Case Related Problems:

Client request:

```
PUT /test-suites/239635/cases/1
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "relatedProblems": [
    {
      "id": 3000000,
      "relationType": "related-to"
    },
    {
      "id": 4000000,
      "relationType": "related-to"
    }
  ]
}
```



```
    ]  
  }
```

**Server response:**

```
HTTP/1.1 201 Created  
Status: 201  
X-API-Version: 1.0
```

**Set Test Suite Case Security:**

**Note:** If Security type is set as either 'Access-Group' or 'Work-Group' then 'name' of the group need to be included with object. If type is 'Person' the 'dsid' of the person should be included.

**Client request:**

```
PUT /test-suites/239635/cases/1  
X-API-Version: 1.0  
Content-Type: application/json; charset=utf-8  
{  
  "security": [  
    {  
      "dsid": 8439,  
      "type": "Person"  
    },  
    {  
      "name": "Radar Developer Group",  
      "type": "Access-Group"  
    }  
  ]  
}
```

**Server response:**

```
HTTP/1.1 201 Created  
Status: 201  
X-API-Version: 1.0
```

### 8.1.5 Get TestSuite Data

**[A] Description**

This API provides a method to get details of test suite.

**[B] Schedule**

Required for version 1.0

**[C] URL Scheme**

```
GET /test-suites/<suite-ID>
```

**[D] Response Parameters**

Parameter	Description	Data Type	Default
title	Title of the test suite.	String	Y
component	A Component Object	Component Object.	Y
status	Status of the test suite.	String	Y
priority	Priority of the test suite.	Integer	Y
geography	Region of the test suite.	String	Y
trackName	Track name of the test suite.	String	Y
applicationName	Application name to which test suite belongs.	String	Y
complexity	Complexity of the test suite.	String	Y
category	Category of the test suite	String	Y
changeID	Change ID of the test suite	String	Y
isActive	Test Suite is active or not.	Boolean	Y
isRegression	Test Suite regression is set or not	Boolean	Y
createdAt	Test Suite creation date	ISO 8601 date string	Y
lastModifiedAt	Test Suite last modification date	ISO 8601 date string	Y
approvedAt	Approval date and time of test suite	ISO 8601 Date time String	N
keywords	Keywords attached to test suite will be returned if fetchKeywords parameter is declared as true in URL.	String Array	N
cases	Case attached to test suite will be returned if fetchCaseData parameter is declared as true in URL.	Array of Case object	N
assignee	Assignee of the test suite. Object will contain dsid, firstName, lastName, email and personType attributes.	Person Object	N
author	Author of the test suite. Object will contain dsid, firstName, lastName, email and personType attributes.	Person Object	N
masterData	Master Data of the test suite.	String	N





Parameter	Description	Data Type	Default
suiteID	ID of the test suite.	Integer	Y
suiteCreator	Creator of the test suite. Object will contain dsid, firstName, lastName, email and personType attributes.	Person Object	N
prerequisites	Prerequisites of the test suite.	String	N
relatedProblems	Related Problem of the test suite. Please refer, RelatedProblem Object for GetTestSuite table for detailed description.	Array of Related Problem Objects	N
diagnosis	User entered text in diagnosis will be returned	JSON Array	N
diagnosis.user	User entered text in diagnosis will be returned	JSON Array	N
diagnosis.history	change history of the test will be returned	JSON Array	N
diagnosis.all	return both types of user entered and history changes	JSON Array	N

#### Diagnosis Object

Parameter	Description	Data Type
text	The diagnosis text	String
addedBy	A person object with firstName, lastName and email	A Person Object
addedAt	The date /time the diagnosis entry was added	A Date/Time

#### Case Object

Key	Description	Data Type
caseID	ID of the case	Integer
priority	Priority of the case	Integer
isActiveFlag	Flag to indicate whether case is active.	Boolean
expectedTime	expected time of finishing the case	String



Key	Description	Data Type
expectedResult	expected result of the case	String
instructions	instruction of the case	String
data	data of the case	String
caseNumber	case number of the case	Integer
summary	summary of the case	String
title	title of the case	String
lastModifiedAt	Last modification date of case	ISO 8601 Datetime string
createdAt	Creation date of case	ISO 8601 Datetime string

#### RelatedProblem Object for GetTestSuite

Key	Description	Data Type
id	ID of the related problem.	Integer
title	Title of the related problem.	String
relationType	Relation of the problem to the test suite.	String
component	Component object containing name and version	Component Object
state	State of the related problem.	String
assignee	Assignee of the related problem.	String
caseNumber	Case Number of the related problem.	Integer or Null

#### [E] Examples

##### Get Test Suite Data:

Client request:

```
GET /test-suites/239635
X-API-Version: 1.0
X-Fields-Requested:
keywords,cases,status,complexity,applicationName,regression,author,track
Name,masterData,changeID,suiteID,lastModifiedAt,suiteCreator,title,catego
ry,geography,prerequisites,relatedProblems,component,priority,createdAt,
isActive
```



Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "keywords": [
    {
      "id": 96288,
      "name": "Test"
    },
    {
      "id": 93811,
      "name": "test"
    }
  ],
  "cases": [
    {
      "lastModifiedAt": "2013-12-16T12:39:50+0000",
      "createdAt": "2013-10-01T05:23:50+0000",
      "title": "test case 1SADFDSA",
      "instructions": "instruction",
      "isActiveFlag": false,
      "expectedResult": "expected result",
      "caseID": 9749000,
      "expectedTime": "0000:59:59",
      "priority": 1,
      "data": "data\n\nSDFADSFADSDFA\ndfsaasdfsFADSDFDASFDAS",
      "caseNumber": 1,
      "summary": "summary"
    }
  ],
  "status": "In Progress",
  "complexity": "Low",
  "applicationName": "BigSlick",
  "isRegression": true,
  "author": {
    "lastName": "K",
    "email": "rakesh_k@apple.com",
    "type": "External",
    "firstName": "Rakesh",
    "dsid": 1141936992
  },
  "trackName": "Admin",
  "masterData": null,
  "changeID": "Radar",
  "suiteID": 418392,
  "lastModifiedAt": "2013-02-19T12:03:00+0530",
  "suiteCreator": {
    "lastName": "K",
    "email": "rakesh_k@apple.com",
    "type": "External",
    "firstName": "Rakesh",
```



```

        "dsid": 1141936992
    },
    "title": "testing Radar WS",
    "category": "Admin",
    "geography": "ALAC",
    "prerequisites": null,
    "relatedProblems": [
        {
            "id": 1002305,
            "title": "test ing",
            "relationType": "related to",
            "componentName": "Alaska",
            "state": "Closed",
            "assignee": "Simpson, Bart",
            "componentVersion": "1.0",
            "caseNumber": 1
        }
    ],
    "component": {
        "name": "Radar",
        "version": "Dev-Automation"
    },
    "priority": 3,
    "createdAt": "2012-08-25T13:32:00+0530",
    "isActive": true
}

```

#### Client request with test suite diagnosis:

```

GET /test-suites/239635
X-API-Version: 1.4
X-Fields-Requested: suiteID,title,diagnosis

```

#### Server response:

```

HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.4
Content-Type: application/json; charset=utf-8

{
  "suiteID": 567425,
  "title": "Test Problem New Title70503",
  "diagnosis": [
    {
      "text": "test",
      "addedAt": "2013-12-16T09:24:33+0000",
      "addedBy": {
        "email": "radartester01@gmail.com",
        "firstName": "Radar",
        "lastName": "Tester1"
      }
    }
  ]
}

```



### 8.1.6 Remove TestSuite Case

#### [A] Description

This API provides method to remove a case from TestSuite.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
DELETE /test-suites/<suite-ID>/cases/<case-number>
```

#### [D] Examples

##### **Remove TestSuite Case:**

The Request URL contains the Test Suite ID and case number of the case which has to be removed from the test.

Client request:

```
DELETE /test-suites/239635/cases/1
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```

### 8.1.7 Find Test Suite

#### [A] Description

This API provides method to find test suite. It supports operators for different attributes as supported by FindProblem API.

The default number of rows returned is 2000. You may increase the number of rows returned by using the “X-rowlimit” header. Note that a large rowlimit value with too broadly defined query criteria can lead to a connection timeout. If this happens, focus your criteria more narrowly.

To get only selected fields in Response, X-Fields-Requested header need to be used with required attributes.



The domain of search values that can be specified for an attribute is dependent on the attribute's data type. The following table describes the list of values that can be specified without the use of operators.

Data Type	Possible Values
Integer	An integer or list of integers. With a list, the result will be problems that match ANY of the supplied values.
Boolean	true or false
Date/time	An ISO 8601 date-time string or date string, or a list of date strings. When a date string is supplied, the search ranges from 00:00 to 24:00 GMT hours on the given date. With a list, the result will be problems that match ANY of the supplied dates.
String	An exact string or list of exact strings to match. With a list, the result will be problems that match ANY of the supplied values.
Enumerated String	One of the enumerated string values, or a list of such values. With a list, the result will be problems that match ANY of the supplied values.
Component	An component specified as an object with "name", "version" and optional "includeSubcomponents" Key.
Component Bundle	An component bundle specified as a string name. If a private and global component bundle both have the same name, the private bundle will be used.
Person	A person specified as an integer DSID, or a list of DSIDs. With a list, the result will be problems that match ANY of the supplied values.
Keyword	A keyword specified as a string name or an integer ID, or a list of such values. Since keywords are a collection attribute, with a list, the result will be problems that match ALL of the supplied values.

### Attribute Operators

Instead of a field value or list of field values, a search can be specified as an object whose Key are arguments. The list of possible arguments are:

Operators	Description
eq	Equality: this is the same as not using an operator.
neq	Not equal
gt	Greater than
gte	Greater than or equal
lt	Less than



Operators	Description
lte	Less than or equal
any	The attribute being searched must contain ANY of the values in the list supplied.
none	The attribute being searched must contain NONE of the values in the list supplied.
like	The attribute being searched must match a wildcard search, with a trailing "%" as the operator.

This table lists the operators that apply to each data type:

Data Type	Supported Operators
Integer	eq, neq, gt, gte, lt, lte, any, none
Date/time	eq, neq, gt, gte, lt, lte, any, none
String	eq, neq any, none, like
Enumerated String	eq, neq, gt, gte, lt, lte, any, none (Ordering is specified by the list of enumerated values, not alphabetically. <a href="#">See 10.1 Get Field Enumeration</a> )
Component	eq, neq, any, none
ComponentBundle	eq, neq, any, none
Person	eq, neq, any, none
Keyword	eq, neq, any, none, all

## [B] Schedule

Required for version 1.0

## [C] URL Scheme

POST /test-suites/find

## [D] Request Parameters

Parameter	Description	Data Type
suiteID	ID of the test suite	Integer
status	Status of test suite	Enumeration String



priority	Priority of test suite	Integer
complexity	Complexity of test suite	Enumeration String
geography	Test region or geography of test suite	Enumeration String
trackName	Track name of test suite	Enumeration String
title	Title of test suite	String
applicationName	Application Name of test suite	String
category	Category of test suite	String
testType	Assigned test type to test suite	String
createdAt	Creation date of test suite	ISO8601 Date String
lastModifiedAt	Last modified date of test suite	ISO8601 Date String
approvedAt	Approved date of test suite	ISO8601 Date String
assignee	DSID of test suite assignee	Person
author	DSID of test suite suite author	Person
lastModifiedBy	DSID of last modified person	Person
createdBy	DSID of test suite creator	Person
approvedBy	DSID of test suite approval	Person
reviewCasesCount	Number of review cases in test suite	Integer
scheduledTestsCount	Number of scheduled test in test suite	Integer
casesCount	Number of cases in test suite	Integer
relatedTestSuitesCount	Number of related test suite	Integer
keyword	Keyword attached to test suite	Keyword
component	A Component Object	Component Object
componentBundle	A Component Bundle Object	ComponentBundle Object
expectedTime	Expected time of test suite. Format should be HHHH:MM:SS	Date String





isRegression	Is test suite regression	Boolean
isActive	Is test suite active	Boolean
additionalWhereClause	This where clause is standard Oracle.SQL	String

### [E] Response Parameters

The response parameter will contain an array of test suite object with below default fields if X-Fields-requested header is not included.

Parameter	Description	Data Type	Default
suiteID	ID of test suite	Integer	Y
title	Title of the test suite	String	Y
applicationName	Application name assigned to test suite	String	Y
category	Category of test suite	String	Y
complexity	Complexity of test suite	String	Y
geography	Test region or geography of test suite	String	Y
label	Label name of label assigned to test suite	String	N
casesCount	Number of case attached to test suite	Integer	N
reviewCasesCount	Number of review case attached to test suite	Integer	N
ccCount	Number of person CC to test suite	Integer	N
relatedTestSuitesCount	Number of related test suite attached	Integer	N
scheduledTestsCount	Number of scheduled test attached	Integer	N
priority	Priority of test suite	Integer	Y
status	Status of test suite	String	Y
totalExpectedTime	Total expected time of test suite. Format will be HHHH:MM:SS	Date String	N



Parameter	Description	Data Type	Default
trackName	Track name of test suite	String	Y
approvedAt	Test suite approval date	ISO 8601 Date String	N
createdAt	Test suite creation date	ISO 8601 Date String	N
lastModifiedAt	Test suite last modification date	ISO 8601 Date String	Y
isActive	Is test suite active	Boolean	Y
isRegression	Is test suite regression	Boolean	Y
approvedBy	The person who approved test suite	person Object	N
createdBy	The person who created test suite	person Object	N
lastModifiedBy	The person who last modified test suite	person Object	N
author	The person who is author of test suite	person Object	N
component	Component to which test suite belongs	component Object	Y
keywordNames	An array containing all the keyword names without its metadata.	Array of String	N
testTypes	Array of assigned test type attached to test suite	Array of String	N

## [F] Examples

### Find Test Suite:

The Request Body can contains any parameters from above request parameter list.

Client request:

```
POST /test-suites/find
X-API-Version: 1.0
X-Fields-Requested:
lastModifiedAt,createdAt,status,suiteID,title,suiteAuthor
Content-Type: application/json; charset=utf-8
{
  "lastModifiedBy": 2002006588,
```



```
    "title": "radar automation"
  }
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "lastModifiedAt": "2011-12-23T16:00:00+00:00",
    "createdAt": "2011-12-22T13:07:00+00:00",
    "status": "In Progress",
    "suiteID": 239635,
    "title": "radar automation sample test",
    "suiteAuthor": {
      "dsid": 102100455,
      "firstName": "Radar",
      "lastName": "Tester1",
      "email": null,
      "type": null
    }
  },
  {
    "lastModifiedAt": "2011-12-24T00:00:00+00:00",
    "createdAt": "2011-12-22T13:07:00+00:00",
    "status": "In Progress",
    "suiteID": 239636,
    "title": "radar automation schedule test",
    "suiteAuthor": {
      "dsid": 102100455,
      "firstName": "Radar",
      "lastName": "Tester1",
      "email": null,
      "type": null
    }
  }
]
```

### Find Test Suite based on approvedAt

```
GET /test-suites/find
X-API-Version: 1.4
X-Fields-Requested: suiteID,title,approvedAt
{
  "approvedAt": "2013-11-23T19:23:50"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.4
Content-Type: application/json; charset=utf-8
```



```
{
  "suiteID": 567425,
  "title": "Test Problem New Title70503",
  "approvedAt": "2013-11-23T19:23:50+0000"
}
```

### 8.1.8 Get Test Suite Enclosures List

#### [A] Description

This API provides a method to fetch enclosures using two different paths: one for attachments, and one for pictures. The enclosure contents can then be retrieved using [2.2 Download Test Suite Enclosure](#).

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
GET /test-suites/<suite_id>/attachments
GET /test-suites/<suite_id>/pictures
GET /test-suites/<suite_id>/cases/<case_id>/attachments
GET /test-suites/<suite_id>/cases/<case_id>/pictures
```

#### [D] Response Attributes

Key	Description	Data Type
fileName	The local path of the enclosure. Note: If the file uploaded contains "/" the slashes are converted to ":"	String
fileSize	The size of the enclosure in bytes.	Integer
addedAt	The date and time that the enclosure was added. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
addedBy	The user who uploaded the enclosure	Person Object

Below fields are added only in response of enclosures (Attachments not pictures).

Key	Description	Data Type
fileId	ID of Enclosure File.	String
createdAt	The date and time that the file was created. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string



Key	Description	Data Type
lastModifiedAt	The date and time that the file was last modified. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
privileges	Privileges of owner and user on file	Object
encodeType	Encoding types used on enclosure file. It will contain all the encoding types used in comma separated string. This encoding is done if file is uploaded through Radar mac client. Web Services does not support encoding of any files at server. Various encode type used are BinHex, Encrypted, Sensitive, AppleSingle, Gzip	String or null

Note about fileName: if a file has been uploaded into a subdirectory, the fileName attribute will include the path to the file, relative to the collection name.

## [E] Examples

### Get list of test suite pictures:

Client request:

```
GET /test-suites/9000000/pictures
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2011-02-14T15:20:00+0000",
    "addedBy": {
      "dsid": 108039134,
      "email": "luke.burton@apple.com",
      "firstName": "Luke",
      "lastName": "Burton",
      "type": "Employee"
    },
    "fileName": "Pasted Picture 1",
    "fileSize": 7400
  },
  {
    "addedAt": "2011-02-14T15:35:00+0000",
    "addedBy": {
      "dsid": 102003482,
      "email": "jfarkas@apple.com",
      "firstName": "Jacob",
      "lastName": "Farkas",

```



```

        "type": "Employee"
    },
    "fileName": "failgail",
    "fileSize": 7400
},
{
    "addedAt": "2011-02-14T16:57:00+0000",
    "addedBy": {
        "dsid": 12102825,
        "email": "mdimaggio@apple.com",
        "firstName": "Matt",
        "lastName": "DiMaggio",
        "type": "Employee"
    },
    "fileName": "Over9000",
    "fileSize": 7400
}
]

```

#### Get list of test suite attachments:

Client request:

```

GET /test-suites/9000000/attachments
X-API-Version: 1.0

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
    {
        "addedAt": "2011-02-14T15:03:30+0000",
        "addedBy": {
            "dsid": 1118580968,
            "email": "radartester01@gmail.com",
            "firstName": "Radar",
            "lastName": "Tester1",
            "type": "Contractor"
        },
        "fileName": "Movies/IMG_0004.MOV",
        "fileSize": 107400,
        "fileId": "RJ1RDR0083761291",
        "createdAt": "2011-02-14T15:03:30+0000",
        "lastModifiedAt": "2011-02-14T15:03:30+0000",
        "encodeType": null,
        "privileges": {
            "owner": {
                "delete": true,
                "write": true,
                "read": true
            }
        }
    },

```



```

        "user": {
            "delete": true,
            "write": true,
            "read": true
        }
    }
}
]

```

#### Get list of test suite attachments (with subfolders):

Client request:

```

GET /test-suites/12311353/attachments
X-API-Version: 1.0

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2013-02-01T17:28:16+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447195",
    "fileName": "WebServicesTestFolder/Screen Shot 2013-02-01 at 8.13.22 AM.png",
    "fileSize": 38888,
    "createdAt": "2013-02-01T17:28:16+0000",
    "lastModifiedAt": "2013-02-01T17:28:16+0000",
    "encodeType": null,
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  },
  {
    "addedAt": "2013-02-01T17:28:16+0000",

```



```

    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447196",
    "fileName": "WebServicesTestFolder/Screen Shot 2013-02-01 at
8.13.20 AM.png",
    "fileSize": 44473,
    "createdAt": "2013-02-01T17:28:16+0000",
    "lastModifiedAt": "2013-02-01T17:28:16+0000",
    "encodeType": "Encrypted,Sensitive",
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  }
}
]

```

#### Get list of test suite attachments (with subfolders that have slashes):

Client request:

```

GET /test-suites/12311353/attachments
X-API-Version: 1.0

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2013-02-01T17:32:18+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447201",

```





```
"fileName": "Web:Services:Folder/Screen Shot 2013-02-01 at
8.13.22 AM.png",
"fileSize": 38888,
"createdAt": "2013-02-01T17:28:16+0000",
"lastModifiedAt": "2013-02-01T17:28:16+0000",
"encodeType": "BinHex,Encrypted,Sensitive",
"privileges": {
  "owner": {
    "delete": true,
    "write": true,
    "read": true
  },
  "user": {
    "delete": true,
    "write": true,
    "read": true
  }
}
},
{
  "addedAt": "2013-02-01T17:32:18+0000",
  "addedBy": {
    "dsid": 1118580968,
    "email": "radartester01@gmail.com",
    "firstName": "Radar",
    "lastName": "Tester1",
    "type": "Contractor"
  },
  "fileId": "RJ1RDR0083447202",
  "fileName": "Web:Services:Folder/Screen Shot 2013-02-01 at
8.13.20 AM.png",
  "fileSize": 44473,
  "createdAt": "2013-02-01T17:32:18+0000",
  "lastModifiedAt": "2013-02-01T17:32:18+0000",
  "encodeType": "BinHex,Encrypted",
  "privileges": {
    "owner": {
      "delete": true,
      "write": true,
      "read": true
    },
    "user": {
      "delete": true,
      "write": true,
      "read": true
    }
  }
}
}
]
```



### 8.1.9 Download Test Suite Enclosure

#### [A] Description

This API provides a method to download a specific attachment or picture, specified by path.

To download all attached file from enclosure in single call, a request need to be send with any filename with .zip extension and "X-Download-All: true" in the header.

Example /test-suites/12174826/attachments/test.zip

All the files will get downloaded as a zip file with the name passed in the request i.e test.zip

If request does not contain the file name with .zip extension ,then an error will be thrown "Please pass file name with .zip extension".

If 'X-Download-All' header is set to false or not pass in request than it will look for filename passed in request in enclosure and download specific file if found otherwise an error will be thrown saying file not attached to problem.

The filename passed in request should be URL encoded before sending, to download proper file from server.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

```
GET /test-suites/<suite_id>/attachments/<enclosure_path>
GET /test-suites/<suite_id>/pictures/<enclosure_path>
GET /test-suites/<suite_id>/cases/<case_id>/attachments/<enclosure_path>
GET /test-suites/<suite_id>/cases/<case_id>/pictures/<enclosure_path>
```

#### [D] Examples

##### Download A Specific Picture:

Client request:

```
GET /test-suites/9000000/pictures/Pasted%20Picture%201
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.4
Content-Type: image/jpeg
Content-Disposition: attachment; filename=Pasted Picture 1
```



<Binary Attachment Content>

#### Download an Attachment from a Subfolder:

Client request:

```
GET /test-suites/9000000/attachments/Movies/IMG_0004.MOV
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: image/jpg
Content-Disposition: attachment; filename=IMG_0004.MOV
<Binary Attachment Content>
```

#### Download all Attachment from Enclosure:

Client request:

```
GET /test-suites/9000000/attachments/9000000.zip
X-Download-All: true
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.4
Status: 200
Content-Type: image/jpg
Content-Disposition: attachment; filename=9000000.zip
<Binary Attachment Content>
```

### 8.1.10 Upload Test Suite Enclosure

#### [A] Description

This API provides a means to upload an enclosure to either the attachments or pictures well. The enclosure path must be specified in the URL, with HTTP headers providing the content type. With attachments (but not pictures), the path can include directories, such as “subfolder/screenshot.png”, and any intermediate directories in the path will be created on the server.

On success, the server responds with 201 Created and no response body.

To Override any attached files , `X-Override-File` header with value as `true` will need to be passed. It will override the existing attached file and upload the new file.



The filename passed in request should be URL encoded in-order to support special characters in filename.

## [B] Schedule

Required for version 1.4

## [C] URL Scheme

```
PUT /test-suites/<suite_id>/attachments/<enclosure_path>
PUT /test-suites/<suite_id>/pictures/<enclosure_path>
PUT /test-suites/<suite_id>/cases/<case_id>/attachments/<enclosure_path>
PUT /test-suites/<suite_id>/cases/<case_id>/pictures/<enclosure_path>
```

## [D] Examples

### Add a picture:

Client request:

```
PUT /test-suites/9000000/pictures/NewImage.jpg
X-API-Version: 1.4
Content-Type: image/jpeg
<Binary Attachment Content>
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.4
```

### Add an attachment:

Client request:

```
PUT /test-suites/9000000/attachments/Movies/IMG_0004.MOV
X-API-Version: 1.4
Content-Type: video/quicktime
<Binary Attachment Content>
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.4
```

### Add an attachment (with X-Override-File):

Client request:

```
PUT /test-suites/9000000/attachments/Movies/IMG_0004.MOV
```



```
X-API-Version: 1.4
X-Override-File: true
Content-Type: video/quicktime
<Binary Attachment Content>
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.4
```

### 8.1.11 Delete Test Suite Enclosure

#### [A] Description

This API provides a method to delete a specific enclosure, specified by path. On success, the server responds with 204 No Content. If the deleted enclosure is part of a subfolder, and the subfolder is empty after deletion, the folder will be deleted.

The filename passed in request should be URL encoded in-order to support special characters in filename.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

```
DELETE /test-suites/<suite_id>/attachments/<enclosure_path>
DELETE /test-suites/<suite_id>/pictures/<enclosure_path>
DELETE /test-suites/<suite_id>/cases/<case_id>/attachments/
<enclosure_path>
DELETE /test-suites/<suite_id>/cases/<case_id>/pictures/<enclosure_path>
```

#### [D] Examples

##### Delete an enclosure:

Client request:

```
DELETE /test-suites/9000000/attachments/Subfolder/Attachment2.txt
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 204 No Content
X-API-Version: 1.4
Status: 204
```



## 8.1.12 Modify Test Suite Enclosure

### [A] Description

This API provides a method to modify the metadata for a specific image or document file -- currently, this just means renaming. The client passes the path to the enclosure, and the request body contains the new enclosure path. On success, the server returns a status of 200 **Success**.

The filename passed in request should be URL encoded in-order to support special characters in filename. The maximum length of filename is 255 characters.

### [B] Schedule

Required for version 1.4

### [C] URL Scheme

```
POST /test-suites/<suite_id>/attachments/<enclosure_path>
POST /test-suites/<suite_id>/pictures/<enclosure_path>
POST /test-suites/<suite_id>/cases/<case_id>/attachments/
<enclosure_path>
POST /test-suites/<suite_id>/cases/<case_id>/pictures/<enclosure_path>
```

### [D] Examples

#### Rename a Picture:

Client request:

```
POST /test-suites/9000000/pictures/OldImage.jpg
X-API-Version: 1.4
Content-Type: application/json; charset=UTF-8
{
  "path": "NewImage.jpg"
}
```

Server response:

```
HTTP/1.1 200 Success
Status: 200
X-API-Version: 1.4
```

#### Rename an Attachment:

Client request:

```
POST /test-suites/9000000/attachments/Movies/IMG_0004.MOV
Content-Type: application/json; charset=UTF-8
X-API-Version: 1.4
{
  "path": "MyMovie.mov"
```



```
}
```

Server response:

```
HTTP/1.1 200 Success
Status: 200
X-API-Version: 1.4
```

## 8.2 Scheduled Test

This Section covers API related to scheduled test.

### 8.2.1 Add Scheduled Test

#### [A] Description

This API provides a method to add a new scheduled test to a existing test suite.

The request can contain any one of datesTrackToMilestone or datesTrackToBuild. If both passed then error message will be shown.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
POST /scheduled-tests
```

#### [D] Request Parameters

Parameter	Description	Data Type
suiteID	ID of the test suite	Integer
component	A Component Object	Component Object
owner	DS ID of the test owner.	Integer
tester	DS ID of the tester.	Integer
scheduledStartDate	Start date of scheduled test.	ISO 8601 date string
scheduledEndDate	End date of scheduled test.	ISO 8601 date string
testCycle	Cycle of the test.	String



Parameter	Description	Data Type
geography	Region of the test.	String
datesTrackToMilestone	Milestone name or ID from which scheduled start and end should be track.	String/Integer
datesTrackToBuild	Build name or ID from which scheduled start and end should be track.	String/Integer
build	ID or name of the build	String/Integer
testConfiguration	Test configuration for scheduled Test	String
scheduleCasesWithPriorities	Priorities of the Test suite cases to be scheduled	Array of Integer
scheduleCasesWithKeywords	Keywords of the Test suite cases to be scheduled	Array of String

### [E] Response Parameters

The response attribute will be same as response attribute section mentioned in 8.2.6 Get Scheduled Test Data. Default response can be override using X-Fields-Requested header.

### [F] Examples

#### Add Scheduled Test:

Client request:

```
POST /scheduled-tests
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "datesTrackToBuild": "6.13.3",
  "component": {"name": "Radar",
  "version": "6.14"},
  "scheduledEndDate": "2011-12-29T16:00:00-08:00",
  "geography": "ALAC",
  "scheduledStartDate": "2011-12-28T13:07:00-08:00",
  "testCycle": "Documentation",
  "owner": 2002021017,
  "suiteID": 239635,
  "tester": 2002006588
}
```

Server response:

```
HTTP/1.1 201 Created
```





```

Status: 201
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "title": "Radar WS Testing",
  "scheduledID": 1315682,
  "scheduledStartDate": "2011-12-28T00:00:00+0000",
  "keywords": [],
  "geography": "ALAC",
  "priority": 5,
  "status": "In Progress",
  "component":
    {
      "name": "Radar",
      "version": "Automation"
    },
  "scheduledEndDate": "2013-12-29T00:00:00+0000",
  "owner":
    {
      "lastName": "Tester1",
      "email": "radartester01@gmail.com",
      "type": "Contractor",
      "firstName": "radar",
      "dsid": 1118580968
    },
  "lastModifiedAt": "2013-04-09T07:38:32+0000",
  "scheduledID": 1315682
}

```

## 8.2.2 Add Case to Scheduled Test

### [A] Description

This API is used to add a case to an existing scheduled test.

### [B] Schedule

Required for version 1.0

### [C] URL Scheme

POST /scheduled-tests/<scheduled-test-ID>

### [D] Request Parameters

Parameter	Description	Data Type
priority	Priority of the case. If omitted priority will default to 5.	Integer



caseNumber	Number of the case. If omitted the new case will be added to the end.	Integer
instructions	Instructions for the case. Maximum length is 4000 characters.	String
data	Case data for testing. Maximum length is 1000000 characters.	String
expectedResult	Expected result of the case testing. Maximum length is 4000 characters.	String
actualResult	Actual result of the case testing. Maximum length is 4000 characters.	String
actualTime	Actual time taken to complete case testing. Format of actualTime is HHHH:MM:SS	Date String
tester	Person ID of tester. If omitted tester will be the logged in person.	Integer
status	Values for caseStatus are: Pass, fail, n/a, no value, blocked	String
title	Title for the case. Maximum length is 240 characters.	String
summary	Summary for the case. Maximum length is 4000 characters.	String
build	ID or name of the build	String/Integer

#### [E] Response Parameters

Parameter	Description	Data Type
caseID	Case ID of newly created scheduled test case	Integer

#### [F] Examples

##### Add Case to Scheduled Test:

Client request:

```
POST /scheduled-tests/551616
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
```



```
"actualResult": "Result Actual",
"actualTime": "10:25:50",
"data": "case data",
"instructions": "case Instruction",
"caseNumber": 1,
"priority": 1,
"status": "Pass",
"title": "title of the case",
"expectedResult": " Expected Result ",
"tester": 2002006588
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
{
  "caseID": 1289308
}
```

### 8.2.3 Remove Scheduled Test Case

#### [A] Description

This API provides method to remove a case from scheduled test.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

```
DELETE /scheduled-tests/<scheduled-test-ID>/<case-number>
```

#### [D] Examples

##### Remove Scheduled Test Case:

The Request URL contains scheduled test ID and case number of the case which has to be removed from the test.

Client request:

```
DELETE /scheduled-tests/551616/1
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.0
```



## 8.2.4 Set Scheduled Test Case Data

### [A] Description

This API provides method to set data to case included in scheduled test.

### [B] Schedule

Required for version 1.0

### [C] URL Scheme

PUT /scheduled-tests/<scheduled-test-ID>/cases/<case-number>

### [D] Request Parameters

Parameter	Description	Data Type
actualResult	Actual result of the case testing. Maximum length is 4000 characters.	String
actualTime	Actual time taken to complete case testing. Format of actualTime is HHHH:MM:SS	Date String
expectedResult	Expected result of the case testing. Maximum length is 4000 characters.	String
data	Case Data for testing. Maximum length is 1000000 characters.	String
instructions	Instructions for the case. Maximum length is 4000 characters.	String
priority	Priority of the case.	Integer
status	Values for case Status are: Pass, fail, n/a, no value, blocked	String
tester	Person ID of the tester.	Integer
title	Title of the case. Maximum length is 240 characters.	String
build	ID or name of the build	String/Integer
summary	Summary of the case. Maximum length is 4000 characters.	String
keywords	Array of keyword name or keywordID	Array of String/Integer



Parameter	Description	Data Type
relatedProblems	Array of related problem object	Array of Related Problem
security	Array of security object	Array of security object

#### Related Problem Object

Parameter	Description	Data Type
id	ProblemID which need to be related to case	Integer
relationType	Relation type of related problem	String

#### Security Object

Parameter	Description	Data Type
type	Security type which need to added. Type can be either of Person, Work-Group or Access-Group.	String
name	Name of Group. This field should be included if type is either of Work-Group or Access-Group.	String
dsid	DSID of person. This field should be included if type is Person.	Integer

#### [E] Examples

##### Set Scheduled Test Case Data:

Client request:

```
PUT /scheduled-tests/551616/1
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "actualResult": "Result Actual",
  "actualTime": "10:25:50",
  "data": "case data",
  "instructions": "case Instruction",
  "priority": 1,
  "status": "Fail",
  "expectedResult": " Expected Result ",
  "tester": 2002006588,
  "title": "title of the case"
}
```



Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### Set Scheduled Test Case Related Problems:

Client request:

```
PUT /scheduled-tests/239635/cases/1
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "relatedProblems": [
    {
      "id": 3000000,
      "relationType": "related-to"
    },
    {
      "id": 4000000,
      "relationType": "related-to"
    }
  ]
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

#### Set Scheduled Test Case Security:

**Note:** If Security type is set as either 'Access-Group' or 'Work-Group' then 'name' of the group need to be included with object. If type is 'Person' the 'dsid' of the person should be included.

Client request:

```
PUT /scheduled-tests/239635/cases/1
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "security": [
    {
      "dsid": 8439,
      "type": "Person"
    },
    {
      "name": "Radar Developer Group",
      "type": "Access-Group"
    }
  ]
}
```



```
}
```

**Server response:**

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

**8.2.5 Find Scheduled Test****[A] Description**

This API provides method to find scheduled test. It supports operators for different attributes as supported by FindProblem API.

The default number of rows returned is 2000. You may increase the number of rows returned by using the “x-rowlimit” header. Note that a large rowlimit value with too broadly defined query criteria can lead to a connection timeout. If this happens, focus your criteria more narrowly.

To get only selected fields in Response, X-Fields-Requested header need to be used with required attributes.

The domain of search values that can be specified for an attribute is dependent on the attribute’s data type. The following table describes the list of values that can be specified without the use of operators.

Data Type	Possible Values
Integer	An integer or list of integers. With a list, the result will be problems that match ANY of the supplied values.
Boolean	true or false
Date/time	An ISO 8601 date-time string or date string, or a list of date strings. When a date string is supplied, the search ranges from 00:00 to 24:00 GMT hours on the given date. With a list, the result will be problems that match ANY of the supplied dates.
String	An exact string or list of exact strings to match. With a list, the result will be problems that match ANY of the supplied values.
Enumerated String	One of the enumerated string values, or a list of such values. With a list, the result will be problems that match ANY of the supplied values.
Component	An component specified as an object with “name”, “version” and optional “includeSubcomponents” Key.
Component Bundle	An component bundle specified as a string name. If a private and global component bundle both have the same name, the private bundle will be used.



Data Type	Possible Values
Person	A person specified as an integer DSID, or a list of DSIDs. With a list, the result will be problems that match ANY of the supplied values.
Keyword	A keyword specified as a string name or an integer ID, or a list of such values. Since keywords are a collection attribute, with a list, the result will be problems that match ALL of the supplied values.

### Attribute Operators

Instead of a field value or list of field values, a search can be specified as an object whose Key are arguments. The list of possible arguments are:

Operators	Description
eq	Equality: this is the same as not using an operator.
neq	Not equal
gt	Greater than
gte	Greater than or equal
lt	Less than
lte	Less than or equal
any	The attribute being searched must contain ANY of the values in the list supplied.
none	The attribute being searched must contain NONE of the values in the list supplied.
all	The attribute being searched must contain ALL of the values in the list supplied.
like	The attribute being searched must match a wildcard search, with a trailing "%" as the operator.

This table lists the operators that apply to each data type:

Data Type	Supported Operators
Integer	eq, neq, gt, gte, lt, lte, any, none
Date/time	eq, neq, gt, gte, lt, lte, any, none
String	eq, neq any, none, like





Data Type	Supported Operators
Enumerated String	eq, neq, gt, gte, lt, lte, any, none (Ordering is specified by the list of enumerated values, not alphabetically. <a href="#">See 10.1 Get Field Enumeration</a> )
Component	eq, neq, any, none
ComponentBundle	eq, neq, any, none
Person	eq, neq, any, none
Keyword	eq, neq, any, none, all

### [B] Schedule

Required for version 1.0

### [C] URL Scheme

POST /scheduled-tests/find

### [D] Request Parameters

Parameter	Description	Data Type
scheduledID	ID of the scheduled test	Integer
suiteID	ID of the test suite	Integer
status	Status of scheduled test	Enumeration String
priority	Priority of scheduled test	Integer
complexity	Complexity of scheduled test	Enumeration String
geography	Test region or geography of scheduled test	Enumeration String
trackName	Track name of scheduled test	Enumeration String
title	Title of scheduled test	String
applicationName	Application Name of scheduled test	String
category	Category of scheduled test	String
build	Name of build attached to scheduled test	String



suiteTitle	Title of test suite	String
datesTrackToBuild	Scheduled start and end date tracks to build	String
datesTrackToMilestone	Scheduled start and end date tracks to milestone	String
testCycle	Scheduled test name or cycle of test	String
createdAt	Creation date of schedule test	ISO8601 Date String
lastModifiedAt	Last modified date of scheduled test	ISO8601 Date String
scheduledStartDate	Start date of scheduled test	ISO8601 Date String
scheduledEndDate	End date of scheduled test	ISO8601 Date String
owner	DSID of scheduled test owner	Person
currentTester	DSID of scheduled test current tester	Person
tester	DSID of scheduled test tester	Person
lastModifiedBy	DSID of scheduled test last modified person	Person
author	DSID of scheduled test author	Person
failCaseCount	Number of failed cases in scheduled test	Integer
problemID	ProblemID to which scheduled test belong	Integer
keyword	ID or name of keyword attached to scheduled test	Keyword
component	Component object containing name and version and optional includeSubcomponents	Component Object
componentBundle	ComponentBundle object containing either id or name of bundle	ComponentBundle Object
isOneTestCaseFailed	Is anyone of case failed in scheduled test	Boolean
completedCasePercent	Percent of completed cases	Integer



passedCasePercent	Percent of passed cases	Integer
failedCasePercent	Percent of failed cases	Integer
blockedCasePercent	Percent of blocked cases	Integer
naCasePercent	Percent of N / A cases	Integer
noValueCasePercent	Percent of No Value cases	Integer
actualTimeAllCases	Total actual time of all cases. Time format HHHH:MM:SS	Date String
actualTimeNoValueCases	Actual time of No Value cases. Time format HHHH:MM:SS	Date String
actualTimePassedCases	Actual time of passed cases. Time format HHHH:MM:SS	Date String
actualTimeFailedCases	Actual time of failed cases. Time format HHHH:MM:SS	Date String
actualTimeBlockedCases	Actual time of blocked cases. Time format HHHH:MM:SS	Date String
actualTimeNACases	Actual time of N / A cases. Time format HHHH:MM:SS	Date String
expectedTimeAllCases	Expected time of all cases. Time format HHHH:MM:SS	Date String
expectedTimeNoValueCases	Expected time of No Value cases. Time format HHHH:MM:SS	Date String
expectedTimePassedCases	Expected time of passed cases. Time format HHHH:MM:SS	Date String
expectedTimeFailedCases	Expected time of failed cases. Time format HHHH:MM:SS	Date String
expectedTimeBlockedCases	Expected time of blocked cases. Time format HHHH:MM:SS	Date String
expectedTimeNACases	Expected time of N / A cases. Time format HHHH:MM:SS	Date String
additionalWhereClause	This where clause is standard Oracle.SQL	String



## [E] Response Parameters

The response parameter will contain an array of scheduled test object with below fields.

Parameter	Description	Data Type	Default
scheduledID	ID of scheduled test	Integer	Y
suiteID	ID of test suite	Integer	Y
title	Title of the scheduled test	String	Y
status	status of the test	String	Y
component	Component object containing name and version	Component Object	Y
author	Scheduled test author person object	Person Object	N
currentTester	Scheduled test currentTester person object	Person Object	Y
lastModifiedBy	Scheduled test lastModifiedBy person object	Person Object	N
owner	Scheduled test owner person object	Person Object	Y
lastModifiedAt	Last modification date of scheduled test	ISO8601 Date String	Y
createdAt	Creation date of scheduled test	ISO8601 Date String	Y
scheduledStartDate	Scheduled start date	ISO8601 Date String	Y
scheduledEndDate	Scheduled end date	ISO8601 Date String	Y
keywords	Array of keyword names attached to scheduled test	Array of String	Y
applicationName	Application name of scheduled test	String	Y
geography	Geography or test region of scheduled test	String	Y



Parameter	Description	Data Type	Default
priority	Priority of scheduled test	Integer	Y
complexity	Complexity of scheduled test	String	Y
testCycle	Scheduled test name or test cycle	String	Y
category	Business process category of scheduled test	String	Y
trackName	Track name of scheduled test	String	Y
build	Name of build attached to scheduled test	String	N
datesTrackToBuildMilestone	Name of milestone or build from which schedule start and end date is tracked	String	N
failCaseCount	Number of failed cases in scheduled test	Integer	N
blockedCaseCount	Number of blocked cases in scheduled test	Integer	N
naCaseCount	Number of N / A cases in scheduled test	Integer	N
noValueCaseCount	Number of No Value cases in scheduled test	Integer	N
passCaseCount	Number of Passed cases in scheduled test	Integer	N
totalCaseCount	Number of Total cases in scheduled test	Integer	N
ccCount	Number of cced person in scheduled test	Integer	N
label	Name of the label to which scheduled test assigned	String	N
relatedProblemCount	Number of related problem to scheduled test	Integer	N
completedCasePercent	Percentage of completed cases in scheduled test	String	N
passedCasePercent	Percentage of passed cases in scheduled test	String	N
failedCasePercent	Percentage of failed cases in scheduled test	String	N
blockedCasePercent	Percentage of blocked cases in scheduled test	String	N



Parameter	Description	Data Type	Default
naCasePercent	Percentage of N/ A cases in scheduled test	String	N
noValueCasePercent	Percentage of No Value cases in scheduled test	String	N
timeCompletedPercent	Percentage of time completed in scheduled test	String	N
actualTimeNoValueCases	Actual time of No Value cases in scheduled test. Format of response is HHHH:MM:SS	String	N
actualTimePassedCases	Actual time of passed cases in scheduled test. Format of response is HHHH:MM:SS	String	N
actualTimeFailedCases	Actual time of failed cases in scheduled test. Format of response is HHHH:MM:SS	String	N
actualTimeBlockedCases	Actual time of blocked cases in scheduled test. Format of response is HHHH:MM:SS	String	N
actualTimeNACases	Actual time of N/ A cases in scheduled test. Format of response is HHHH:MM:SS	String	N
actualTimeAllCases	Actual time of all cases in scheduled test. Format of response is HHHH:MM:SS	String	N
expectedTimeAllCases	Expected time of all cases in scheduled test. Format of response is HHHH:MM:SS	String	N
expectedTimeNoValueCases	Expected time of No Value cases in scheduled test. Format of response is HHHH:MM:SS	String	N
expectedTimePassedCases	Expected time of passed cases in scheduled test. Format of response is HHHH:MM:SS	String	N
expectedTimeFailedCases	Expected time of failed cases in scheduled test. Format of response is HHHH:MM:SS	String	N
expectedTimeBlockedCases	Expected time of blocked cases in scheduled test. Format of response is HHHH:MM:SS	String	N
expectedTimeNACases	Expected time of N/ A cases in scheduled test. Format of response is HHHH:MM:SS	String	N

## [F] Examples

### Find Scheduled Test:

The Request Body can contains any parameters from above request parameter list.



Client request:

```
POST /scheduled-tests/find
X-API-Version: 1.0
X-Fields-Requested:
scheduledEndDate,scheduledStartDate,status,suiteID,testID,title
Content-Type: application/json; charset=utf-8
{
  "tester": 2002006588,
  "title": "radar automation"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "scheduledEndDate": "2011-12-23T16:00:00+00:00",
    "scheduledStartDate": "2011-12-22T13:07:00+00:00",
    "status": "Pass",
    "suiteID": 239635,
    "scheduledID": 551616,
    "title": "radar automation sample test"
  },
  {
    "scheduledEndDate": "2011-12-24T00:00:00+00:00",
    "scheduledStartDate": "2011-12-22T13:07:00+00:00",
    "status": "fail",
    "suiteID": 239635,
    "scheduledID": 551617,
    "title": "radar automation schedule test"
  }
]
```

## 8.2.6 Get Scheduled Test Data

### [A] Description

This API provides method to get data of scheduled test. The response will contain only default attributes mentioned in response parameter section. To get the non-default parameter `X-Fields-Requested` header need to be used with the required parameter.

### [B] Schedule

Required for version 1.0

### [C] URL Scheme

```
GET /scheduled-tests/<scheduled-test-ID>
```



## [D] Response Parameters

Parameter	Description	Data Type	Default
scheduledID	ID of scheduled test	Integer	Y
suiteID	ID of test suite	Integer	N
component	Object Containing component name and version	Object	Y
status	Status of the schedule test	String	Y
priority	Priority of the test	Integer	Y
owner	Owner of the test	Person Object	Y
author	Author of test Suite	Person Object	N
currentTester	Current tester of the test	Person Object	N
testCycle	Scheduled test name or test cycle	String	N
testConfiguration	Scheduled test configuration	String	N
masterData	Scheduled test master Data	String	N
title	Title of Scheduled test	String	Y
datesTrackToMilestone	Milestone name from which scheduled start and end date tracked	String	N
datesTrackToBuild	Build name from which scheduled start and end date tracked	String	N
build	Build name attached to test	String	N
buildID	ID of build attached to test	Integer	N
complexity	Complexity of scheduled test	String	N
prerequisites	Prerequisites data of scheduled test	String	N
testRestartInstructions	Scheduled test restart instructions	String	N
scheduledStartDate	Scheduled start date of test	ISO 8601 date string	Y
scheduledEndDate	Scheduled end date of test	ISO 8601 date string	Y
geography	The region to which the test is associated with	String	Y
lastModifiedAt	Date when the test last modified.	ISO 8601 date string	Y





Parameter	Description	Data Type	Default
blockedCaseCount	Number of blocked cases in Test	Integer	N
failCaseCount	Number of fail cases in Test	Integer	N
naCaseCount	Number of N/A cases in Test	Integer	N
noValueCaseCount	Number of No Value cases in Test	Integer	N
passCaseCount	Number of pass cases in Test	Integer	N
totalCompletedCases	Number of total completed cases in Test	Integer	N
totalCaseCount	Number of total cases in Test	Integer	N
statusID	ID of scheduled test status	Integer	N
statusIndex	Index of scheduled test status	Integer	N
keywords	Keywords attached to the scheduled test	Array of objects	Y
cases	Cases associated with the test	Array of case object	N
relatedProblems	Related Problem associated with test	Array of related problem Object	N
diagnosis	User entered text in diagnosis will be returned	JSON Array	N
diagnosis.user	User entered text in diagnosis will be returned	JSON Array	N
diagnosis.history	change history of the test will be returned	JSON Array	N
diagnosis.all	return both types of user entered and history changes	JSON Array	N

#### Diagnosis Object

Parameter	Description	Data Type
text	The diagnosis text	String
addedBy	A person object with firstName, lastName and email	A Person Object



Parameter	Description	Data Type
addedAt	The date /time the diagnosis entry was added	A Date/ Time

#### Keyword Object

Parameter	Description	Data Type
id	ID of keyword	Integer
name	Name of keyword	String

#### Related Problem Object

Parameter	Description	Data Type
id	ID of related problem	Integer
title	Title of keyword	String
relationType	Relation type of related problem	String
component	Component of related problem	Component Object
priority	Priority of related problem	Integer
caseID	ID of scheduled test case	Integer
state	State of related problem	String
caseNumber	Case number of scheduled test case	Integer

#### Case Object

Parameter	Description	Data Type
caseID	ID of test case	Integer
caseNumber	Case number of test case	Integer
title	Title of test case	String
priority	Priority of test case	Integer
status	Status of test case	String



Parameter	Description	Data Type
build	Build of test case	String
buildID	ID of build	Integer
reviewFlag	Review flag test case	Boolean
actualTime	Actual time of test case	String
expectedTime	Expected time of test case	String
data	Data of test case	String
instructions	Instruction of test case	String
expectedResult	Expected result of test case	String
actualResult	Actual result of test case	String
summary	Summary of test case	String
tester	Tester of test case	Person Object
lastModifiedAt	Last modification date of case	ISO 8601 Datetime string
createdAt	Creation date of case	ISO 8601 Datetime string
keywords	Keywords attached to test case	Array of Keyword Object
relatedProblems	Related problems attached to test case	Array of Related problem Object

### [E] Examples

#### Get Scheduled Test Data:

Client request:

```
GET /scheduled-tests/551616
X-API-Version: 1.0
X-Fields-Requested:
build,component,geography,lastModifiedAt,priority,scheduledEndDate,sched
uledStartDate,status,scheduledID
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
```



```
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "build": null,
  "component": {
    "name": "Radar (new bugs)",
    "version": "All"
  },
  "geography": "ALAC",
  "lastModifiedAt": "2011-12-21T16:00:00+0000",
  "priority": 5,
  "scheduledEndDate": "2011-12-23T16:00:00+0000",
  "scheduledStartDate": "2011-12-22T13:07:00+0000",
  "status": "In Progress",
  "scheduledID": 551616
}
```

#### Example of Scheduled test case data:

##### Client request:

```
GET /scheduled-tests/551616
X-API-Version: 1.3
X-Fields-Requested: cases
```

##### Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.3
Content-Type: application/json; charset=utf-8

{
  "cases": [
    {
      "lastModifiedAt": "2013-12-16T12:39:50+0000",
      "createdAt": "2013-10-01T05:23:50+0000",
      "tester": {
        "lastName": "Naik",
        "email": null,
        "type": null,
        "firstName": "Arunraj",
        "dsid": 143677002
      },
      "reviewFlag": false,
      "keywords": [
        {
          "id": 78790,
          "name": "Radar"
        }
      ],
      "instructions": "- Create a problem from schedule test step\n- Try un-relating the problem",
      "status": "N/A",
      "actualResult": null,
      "expectedTime": "0000:00:00",
    }
  ]
}
```



```

        "title": "- Create a problem from schedule test step - Try un-
relating the problem",
        "priority": 2,
        "build": null,
        "data": "Test Data",
        "caseNumber": 1,
        "relatedProblems": [
            {
                "id": 6608298,
                "title": "Test- Created by Iswar Das -",
                "relationType": "related to",
                "component": {
                    "name": "Radar",
                    "version": "Dev-6.11"
                },
                "state": "Closed"
            },
            {
                "actualTime": "0000:00:00",
                "expectedResult": "It should give an error.",
                "caseID": 3314525,
                "summary": null
            }
        ],
        ...more records
    ]

```

#### Example of Multiple Get Scheduled Test Data:

Client request:

```

GET /scheduled-tests/551616,225518
X-API-Version: 1.0
X-Fields-Requested:
build,component,geography,lastModifiedAt,priority,scheduledEndDate,sched
uledStartDate,status,scheduledID

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[ {
    "build": null,
    "component": {
        "name": "Radar (new bugs)",
        "version": "All"
    },
    "geography": "ALAC",
    "lastModifiedAt": "2011-12-21T16:00:00+0000",
    "priority": 5,
    "scheduledEndDate": "2011-12-23T16:00:00+0000",
    "scheduledStartDate": "2011-12-22T13:07:00+0000",
    "status": "In Progress",

```



```

    "scheduledID": 551616
  },
  {
    "build": "Database",
    "component": {
      "name": "Radar (new bugs)",
      "version": "All"
    },
    "geography": "ALAC",
    "lastModifiedAt": "2011-12-21T16:00:00+0000",
    "priority": 5,
    "scheduledEndDate": "2011-12-23T16:00:00+0000",
    "scheduledStartDate": "2011-12-22T13:07:00+0000",
    "status": "In Progress",
    "scheduledID": 225518
  }
]

```

#### Client request with scheduled test diagnosis:

Client request:

```

GET /scheduled-tests/1834912
X-API-Version: 1.4
X-Fields-Requested: scheduledID,diagnosis

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.4
Content-Type: application/json; charset=utf-8

{
  "scheduledID": 1834912,
  "diagnosis": [
    {
      "text": "test",
      "addedAt": "2013-12-16T09:46:02+0000",
      "addedBy": {
        "email": "radartester01@gmail.com",
        "firstName": "Radar",
        "lastName": "Tester1"
      }
    }
  ]
}

```

#### Error message for Scheduled Test ID which does not exists in DB

Client request:

```
GET /scheduled-tests/213123
```

Server response:



```
HTTP/1.1 404 Not Found
Content-Type: application/json; charset=utf-8
{
  "message": "Scheduled Test with ID '213123' does not found in
Database.",
  "title": "Scheduled Test not found",
  "help": "View documentation at http://radar.apple.com/",
  "status": "404 Not Found"
}
```

## 8.2.7 Set Scheduled Test Data

### [A] Description

This API provides method to update the fields of scheduled test.

### [B] Schedule

Required for version 1.0

### [C] URL Scheme

PUT /scheduled-tests/<scheduled-test-ID>

### [D] Request Parameters

Parameter	Description	Data Type
scheduledStartDate	Start Date of Scheduled test	ISO 8601 date string
scheduledEndDate	End Date of Scheduled test	ISO 8601 date string
component	A Component Object	Component Object
keywords	Keywords List of the test	String Array
status	Test Status of the test	String
priority	Test Priority of the test	Integer
testCycle	Test Cycle of the test	String
geography	Test region of scheduled test	String
testConfiguration	Test configuration of the test	String
diagnosis	Diagnosis data of the test	String
masterData	Master Data of the test	String



Parameter	Description	Data Type
title	Title of scheduled test.	String
datesTrackToBuild	Milestone name or ID from which scheduled start and end should be track.	String/Integer
datesTrackToMilestone	Build name or ID from which scheduled start and end should be track.	String/Integer
build	ID or name of the build	String/Integer
complexity	Complexity of scheduled test	String
prerequisites	Prerequisites text of scheduled test	String
testRestartInstructions	Test Restart Instruction of scheduled test	String
labelID	LabelID of the label. To remove label from scheduled test, 'null' should be used.	Integer or null

## [E] Examples

### Set Scheduled Test Data:

The request body will contain only those parameter(s) which should be set.

Client request:

```
PUT /scheduled-tests/551616
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "component": {"name": "Radar",
    "version": "Dev-6.12"},
  "status": "In Progress"
}
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.0
```

## 8.2.8 Get Scheduled Test Enclosures List

### [A] Description

This API provides a method to fetch enclosures using two different paths: one for attachments, and one for pictures. The enclosure contents can then be retrieved using [Download Scheduled Test Enclosure](#).



**[B] Schedule**

Required for version 1.4

**[C] URL Scheme**

```
GET /scheduled-tests/<scheduled-test-ID>/attachments
GET /scheduled-tests/<scheduled-test-ID>/pictures
GET /scheduled-tests/<scheduled-test-ID>/cases/<case_id>/attachments
GET /scheduled-tests/<scheduled-test-ID>/cases/<case_id>/pictures
```

**[D] Response Attributes**

Key	Description	Data Type
fileName	The local path of the enclosure. Note: If the file uploaded contains "/" the slashes are converted to ":"	String
fileSize	The size of the enclosure in bytes.	Integer
addedAt	The date and time that the enclosure was added. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
addedBy	The user who uploaded the enclosure	Object

Below fields are added only in response of enclosures (Attachments not pictures).

Key	Description	Data Type
fileId	ID of Enclosure File.	String
createdAt	The date and time that the file was created. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
lastModifiedAt	The date and time that the file was last modified. (YYYY-MM-DDThh:mm:ssTZD)	ISO 8601 date string
privileges	Privileges of owner and user on file	Object
encodeType	Encoding types used on enclosure file. It will contain all the encoding types used in comma separated string. This encoding is done if file is uploaded through Radar mac client. Web Services does not support encoding of any files at server. Various encode type used are BinHex, Encrypted, Sensitive, AppleSingle, Gzip	String or null

Note about fileName: if a file has been uploaded into a subdirectory, the fileName attribute will include the path to the file, relative to the collection name.



## [E] Examples

### Get list of scheduled test pictures:

Client request:

```
GET /scheduled-tests/9000000/pictures
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.4
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2011-02-14T15:20:00+0000",
    "addedBy": {
      "dsid": 108039134,
      "email": "luke.burton@apple.com",
      "firstName": "Luke",
      "lastName": "Burton",
      "type": "Employee"
    },
    "fileName": "Pasted Picture 1",
    "fileSize": 7400
  },
  {
    "addedAt": "2011-02-14T15:35:00+0000",
    "addedBy": {
      "dsid": 102003482,
      "email": "jfarkas@apple.com",
      "firstName": "Jacob",
      "lastName": "Farkas",
      "type": "Employee"
    },
    "fileName": "failgail",
    "fileSize": 7400
  },
  {
    "addedAt": "2011-02-14T16:57:00+0000",
    "addedBy": {
      "dsid": 12102825,
      "email": "mdimaggio@apple.com",
      "firstName": "Matt",
      "lastName": "DiMaggio",
      "type": "Employee"
    },
    "fileName": "Over9000",
    "fileSize": 7400
  }
]
```



**Get list of scheduled test attachments:**

Client request:

```
GET /scheduled-tests/9000000/attachments
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.4
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2011-02-14T15:03:30+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileName": "Movies/IMG_0004.MOV",
    "fileSize": 107400,
    "fileId": "RJ1RDR0083761291",
    "createdAt": "2011-02-14T15:03:30+0000",
    "lastModifiedAt": "2011-02-14T15:03:30+0000",
    "encodeType": null,
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  }
]
```

**Get list of scheduled test attachments (with subfolders):**

Client request:

```
GET /scheduled-tests/12311353/attachments
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 200 OK
```



```
Status: 200
X-API-Version: 1.4
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2013-02-01T17:28:16+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447195",
    "fileName": "WebServicesTestFolder/Screen Shot 2013-02-01 at
8.13.22 AM.png",
    "fileSize": 38888,
    "createdAt": "2013-02-01T17:28:16+0000",
    "lastModifiedAt": "2013-02-01T17:28:16+0000",
    "encodeType": null,
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  },
  {
    "addedAt": "2013-02-01T17:28:16+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447196",
    "fileName": "WebServicesTestFolder/Screen Shot 2013-02-01 at
8.13.20 AM.png",
    "fileSize": 44473,
    "createdAt": "2013-02-01T17:28:16+0000",
    "lastModifiedAt": "2013-02-01T17:28:16+0000",
    "encodeType": "Encrypted,Sensitive",
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  }
]
```



```

        "user": {
            "delete": true,
            "write": true,
            "read": true
        }
    }
}
]

```

**Get list of scheduled test attachments (with subfolders that have slashes):**

Client request:

```

GET /scheduled-tests/12311353/attachments
X-API-Version: 1.4

```

Server response:

```

HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.4
Content-Type: application/json; charset=utf-8
[
  {
    "addedAt": "2013-02-01T17:32:18+0000",
    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447201",
    "fileName": "Web:Services:Folder/Screen Shot 2013-02-01 at 8.13.22 AM.png",
    "fileSize": 38888,
    "createdAt": "2013-02-01T17:28:16+0000",
    "lastModifiedAt": "2013-02-01T17:28:16+0000",
    "encodeType": "BinHex,Encrypted,Sensitive",
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  },
  {
    "addedAt": "2013-02-01T17:32:18+0000",

```



```

    "addedBy": {
      "dsid": 1118580968,
      "email": "radartester01@gmail.com",
      "firstName": "Radar",
      "lastName": "Tester1",
      "type": "Contractor"
    },
    "fileId": "RJ1RDR0083447202",
    "fileName": "Web:Services:Folder/Screen Shot 2013-02-01 at
8.13.20 AM.png",
    "fileSize": 44473,
    "createdAt": "2013-02-01T17:32:18+0000",
    "lastModifiedAt": "2013-02-01T17:32:18+0000",
    "encodeType": "BinHex,Encrypted",
    "privileges": {
      "owner": {
        "delete": true,
        "write": true,
        "read": true
      },
      "user": {
        "delete": true,
        "write": true,
        "read": true
      }
    }
  }
}
]

```

### 8.2.9 Download Scheduled Test Enclosure

#### [A] Description

This API provides a method to download a specific attachment or picture, specified by path.

To download all attached file from enclosure in single call, a request need to be send with any filename with .zip extension and "X-Download-All: true" in the header.

Example /test-suites/12174826/attachments/test.zip

All the files will get downloaded as a zip file with the name passed in the request i.e test.zip

If request does not contain the file name with .zip extension ,then an error will be thrown "Please pass file name with .zip extension".

If 'X-Download-All' header is set to false or not pass in request than it will look for filename passed in request in enclosure and download specific file if found otherwise an error will be thrown saying file not attached to problem.

The filename passed in request should be URL encoded before sending, to download proper file from server.



## [B] Schedule

Required for version 1.4

## [C] URL Scheme

```
GET /scheduled-tests/<scheduled_test_id>/attachments/<enclosure_path>
GET /scheduled-tests/<scheduled_test_id>/pictures/<enclosure_path>
GET /scheduled-tests/<scheduled_test_id>/cases/<case_id>/attachments/
<enclosure_path>
GET /scheduled-tests/<scheduled_test_id>/cases/<case_id>/pictures/
<enclosure_path>
```

## [D] Examples

### Download A Specific Picture:

Client request:

```
GET /scheduled-tests/9000000/pictures/Pasted%20Picture%201
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.4
Content-Type: image/jpeg
Content-Disposition: attachment; filename=Pasted Picture 1
<Binary Attachment Content>
```

### Download an Attachment from a Subfolder:

Client request:

```
GET /scheduled-tests/9000000/attachments/Movies/IMG_0004.MOV
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.0
Status: 200
Content-Type: image/jpeg
Content-Disposition: attachment; filename=IMG_0004.MOV
<Binary Attachment Content>
```

### Download all Attachment from Enclosure:

Client request:



```
GET /scheduled-tests/9000000/attachments/9000000.zip
X-Download-All: true
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 200 OK
X-API-Version: 1.4
Status: 200
Content-Type: image/jpg
Content-Disposition: attachment; filename=9000000.zip
<Binary Attachment Content>
```

## 8.2.10 Upload Scheduled Test Enclosure

### [A] Description

This API provides a means to upload an enclosure to either the attachments or pictures well. The enclosure path must be specified in the URL, with HTTP headers providing the content type. With attachments (but not pictures), the path can include directories, such as “subfolder/screenshot.png”, and any intermediate directories in the path will be created on the server.

On success, the server responds with 201 Created and no response body.

To Override any attached files , `X-Override-File` header with value as `true` will need to be passed. It will override the existing attached file and upload the new file.

The filename passed in request should be URL encoded in-order to support special characters in filename.

### [B] Schedule

Required for version 1.4

### [C] URL Scheme

```
PUT /scheduled-tests/<scheduled_test_id>/attachments/<enclosure_path>
PUT /scheduled-tests/<scheduled_test_id>/pictures/<enclosure_path>
PUT /scheduled-tests/<scheduled_test_id>/cases/<case_id>/attachments/
  <enclosure_path>
PUT /scheduled-tests/<scheduled_test_id>/cases/<case_id>/pictures/
  <enclosure_path>
```

### [D] Examples

**Add a picture:**

Client request:





```
PUT /scheduled-tests/9000000/pictures/NewImage.jpg
X-API-Version: 1.4
Content-Type: image/jpg
<Binary Attachment Content>
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.4
```

#### **Add an attachment:**

Client request:

```
PUT /scheduled-tests/9000000/attachments/Movies/IMG_0004.MOV
X-API-Version: 1.4
Content-Type: video/quicktime
<Binary Attachment Content>
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.4
```

#### **Add an attachment (with X-Override-File):**

Client request:

```
PUT /scheduled-tests/9000000/attachments/Movies/IMG_0004.MOV
X-API-Version: 1.4
X-Override-File: true
Content-Type: video/quicktime
<Binary Attachment Content>
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.4
```

### **8.2.11 Delete Scheduled Test Enclosure**

#### **[A] Description**

This API provides a method to delete a specific enclosure, specified by path. On success, the server responds with 204 No Content. If the deleted enclosure is part of a subfolder, and the subfolder is empty after deletion, the folder will be deleted.



The filename passed in request should be URL encoded in-order to support special characters in filename.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

```
DELETE /scheduled-tests/<scheduled_test_id>/attachments/<enclosure_path>
DELETE /scheduled-tests/<scheduled_test_id>/pictures/<enclosure_path>
DELETE /scheduled-tests/<scheduled_test_id>/cases/<case_id>/attachments/
<enclosure_path>
DELETE /scheduled-tests/<scheduled_test_id>/cases/<case_id>/pictures/
<enclosure_path>
```

#### [D] Examples

##### Delete an enclosure:

Client request:

```
DELETE /scheduled-tests/9000000/attachments/Subfolder/Attachment2.txt
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 204 No Content
X-API-Version: 1.4
Status: 204
```

## 8.2.12 Modify Scheduled Test Enclosure

#### [A] Description

This API provides a method to modify the metadata for a specific image or document file -- currently, this just means renaming. The client passes the path to the enclosure, and the request body contains the new enclosure path. On success, the server returns a status of 200 **Success**.

The filename passed in request should be URL encoded in-order to support special characters in filename. The maximum length of filename is 255 characters.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

```
POST /scheduled-tests/<scheduled_test_id>/attachments/<enclosure_path>
```



```
POST /scheduled-tests/<scheduled_test_id>/pictures/<enclosure_path>
POST /scheduled-tests/<scheduled_test_id>/cases/<case_id>/attachments/
<enclosure_path>
POST /scheduled-tests/<scheduled_test_id>/cases/<case_id>/pictures/
<enclosure_path>
```

#### **[D] Examples**

##### **Rename a Picture:**

Client request:

```
POST /scheduled-tests/9000000/pictures/OldImage.jpg
X-API-Version: 1.4
Content-Type: application/json;charset=UTF-8
{
  "path": "NewImage.jpg"
}
```

Server response:

```
HTTP/1.1 200 Success
Status: 200
X-API-Version: 1.4
```

##### **Rename an Attachment:**

Client request:

```
POST /scheduled-tests/9000000/attachments/Movies/IMG_0004.MOV
Content-Type: application/json;charset=UTF-8
X-API-Version: 1.4
{
  "path": "MyMovie.mov"
}
```

Server response:

```
HTTP/1.1 200 Success
Status: 200
X-API-Version: 1.4
```



## 9. RAW SQL EXECUTION

This Section of APIs provide method to execute a raw SQL statement.

### 9.1 LookUp SQL

#### [A] Description

This API provides method to execute any SQL statement.

The default number of rows returned is 2000. You may increase the number of rows returned by using the “rowlimit” header. Note that a large rowlimit value with too broadly defined query criteria can lead to a connection timeout. If this happens, focus your criteria more narrowly.

#### [B] Schedule

Required for version 1.0

#### [C] URL Scheme

POST /SQL

#### [D] Request Parameters

Parameter	Description	Data Type
statement	SQL Statement to be executed	String

#### [E] Response Parameters

Parameter	Description	Data Type
results	Result array containing object with requested fields in request query	Array of Objects
resultRows	Number of rows fetch for request query	Integer

#### [E] Examples

##### LookUp SQL

The Response will be the JSON String containing result of executed SQL statement send as request. Range parameter in request header will contain the record range to be returned in the response.

Client request:

```
POST /SQL
X-API-Version: 1.0
```



```
Range: records=1-5
Content-Type: application/json; charset=utf-8
{
  "statement": "select PROBLEMID, ORIGAN_PERSONID, ASSIGNED_PERSONID
from FindBundleProblem where PROBLEMID =10025214"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
{
  "results":
  [{
    "ASSIGNED_PERSONID": 1118581234,
    "ORIGAN_PERSONID": 1118581234,
    "PROBLEMID": 10025214
  }],
  "resultRows":1
}
```



## 10. ENUMERATIONS

These APIs provide a means to look up enumerations (e.g. problem states) so that they can be used for sorting, displaying in popup menus, etc.

### 10.1 Get Field Enumeration

#### [A] Description

This API is used to fetch the authoritative list of enumerations for an enumerated field. Enumerations are used for sorting result sets and displaying popup menus for user selection. This API allows clients to avoid using enumeration indexes, which are subject to change over time.

The full list of enumerated fields is:

- problems / states
- problems / substates
- problems / resolutions
- problems / classifications
- problems / reproducibilities
- problems / fix-orders
- problems / priorities
- problems / diagnosis / types
- problems / relations / types
- problems / security-list / statuses
- people / types
- components / milestones / privileges
- test-suites / status
- test-suites / complexity
- scheduled-test / status
- scheduled-test / cases / status
- problems / product-security / types
- problems / product-security / colors
- problems / product-security / attack-vectors
- problems / product-security / exploitabilities
- problems / product-security / authentications
- problems / product-security / impacts
- problems / product-security / assets
- problems / product-security / user-bases
- problems / product-security / complexities
- problems / product-security / visibilities
- components / person-roles
- security / privileges
- group-types
- problems / assignees / types
- problems / mask / types
- recent-queries / types
- shared-reports / subscribers / types
- shared-reports / subscribers / permission
- recent-queries / executed-as

**[B] Schedule**

Required for version 1.0.

**[C] URL Scheme**

```
GET /enumerations/<field-path>
```

**[D] Response Attributes**

The response is an array of strings. The ordering of results indicates the intended field ordering.

**[E] Examples****Get Enumeration of Problem States**

Client request:

```
GET /enumerations/problems/states
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
  "Analyze",
  "Integrate",
  "Build",
  "Verify",
  "Closed"
]
```

**Get Enumeration of Problem Sub States**

Client request:

```
GET /enumerations/problems/substates
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
```



```
"Screen",  
"Investigate",  
"Fix",  
"Review",  
"Prepare",  
"Nominate"  
]
```

### Get Enumeration of Problem Resolution

Client request:

```
GET /enumerations/problems/resolutions  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
X-API-Version: 1.0  
Content-Type: application/json;charset=UTF-8  
[  
  "Unresolved",  
  "Software Changed",  
  "Documentation Changed",  
  "Hardware Changed",  
  "Configuration Changed",  
  "Feature Removed",  
  "Duplicate",  
  "Cannot Reproduce",  
  "Behaves Correctly",  
  "Not To Be Fixed",  
  "3rd Party To Resolve",  
  "Firmware Changed",  
  "Vendor Disqualified",  
  "Process Changed",  
  "Insufficient Information",  
  "Item Completed",  
  "Not Applicable"  
]
```

### Get Enumeration of Problem Classifications

Client request:

```
GET /enumerations/problems/classifications  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
X-API-Version: 1.0  
Content-Type: application/json;charset=UTF-8
```





```
[
  "Security",
  "Crash/Hang/Data Loss",
  "Power",
  "Performance",
  "UI/Usability",
  "Serious Bug",
  "Other Bug",
  "Feature (New)",
  "Enhancement",
  "Task"
]
```

### Get Enumeration of Problem Reproducibilities

Client request:

```
GET /enumerations/problems/reproducibilities
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "Always",
  "Sometimes",
  "Rarely",
  "Unable",
  "I Didn't Try",
  "Not Applicable"
]
```

### Get Enumeration of Problem Fix Orders

Client request:

```
GET /enumerations/problems/fix-orders
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "1",
  "2",
  "3",
  "4",
  "5",
]
```



```
    "6 - Not Set"  
  ]
```

### Get Enumeration of Problem Priority

Client request:

```
GET /enumerations/problems/priorities  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
X-API-Version: 1.0  
Content-Type: application/json; charset=UTF-8  
[  
  "1 - Show stopper",  
  "2 - Expected",  
  "3 - Important",  
  "4 - Nice to have",  
  "5 - Not Set",  
  "6 - Investigate"  
]
```

### Get Enumeration of Problem Diagnosis Types

Client request:

```
GET /enumerations/problems/diagnosis/types  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
X-API-Version: 1.0  
Content-Type: application/json; charset=UTF-8  
[  
  "all",  
  "history",  
  "user"  
]
```

### Get Enumeration of Problem States

Client request:

```
GET /enumerations/problems/relations/types  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
```



```
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
  "related-to",
  "blocked-by",
  "blocking",
  "parent-of",
  "subtask-of",
  "clone-of",
  "cloned-to",
  "duplicate-of",
  "original-of"
]
```

### Get Enumeration of Problem States

Client request:

```
GET /enumerations/problems/state
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
  "Analyze",
  "Integrate",
  "Build",
  "Verify",
  "Closed"
]
```

### Get Enumeration of Problem Security type

Client request:

```
GET /enumerations/problems/security-list/statuses
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
  "Person",
  "Work-group",
  "Access-group"
]
```



## Get Enumeration of People Types

Client request:

```
GET /enumerations/people/types
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "Employee",
  "Contractor",
  "External",
  "No Access"
]
```

## Get Enumeration of Component Milestone Privileges

Client request:

```
GET /enumerations/components/milestones/privileges
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "Read Only",
  "Assignable",
  "Reassignable(Tree-limited)",
  "Reassignable(Full)"
]
```

## Get Enumeration of Test Suite Status

Client request:

```
GET /enumerations/test-suites/status
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
```



```
[
    "Waiting for Data",
    "In Progress",
    "Draft Review",
    "Final Review",
    "Approved"
    "Rejected"
    "Ready for test to be scheduled"
    "Test scheduled"
    "SME Complete"
    "Ready For Automation"
    "Retest"
    "Duplicate"
    "Cancelled"
]
```

### Get Enumeration of Test Suite Complexity

Client request:

```
GET /enumerations/test-suites/complexity
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
    "Low,,",
    "Medium",
    "High"
]
```

### Get Enumeration of Scheduled Test Status

Client request:

```
GET /enumerations/scheduled-test/status
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
    "Not Started",
    "In Progress",
    "Failed",
    "Complete",
    "Cancelled",
]
```



```
"Critical Cases Complete",  
"Blocked"  
]
```

#### Get Enumeration of Scheduled Test Case Status

Client request:

```
GET /enumerations/scheduled-test/cases/status  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
X-API-Version: 1.0  
Content-Type: application/json;charset=UTF-8  
[  
  "Pass",  
  "Fail",  
  "N/A",  
  "Blocked"  
]
```

#### Get Enumeration of Problem Product Security Type

Client request:

```
GET /enumerations/problems/product-security/types  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
X-API-Version: 1.0  
Content-Type: application/json;charset=UTF-8  
[  
  "Exposure",  
  "Mitigation",  
  "None"  
]
```

#### Get Enumeration of Problem Product Security Color

Client request:

```
GET /enumerations/problems/product-security/colors  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200
```



```
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "Red",
  "Orange",
  "Yellow",
  "Blue"
]
```

### Get Enumeration of Problem Product Security AttackVector

Client request:

```
GET /enumerations/problems/product-security/attack-vectors
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "API",
  "Local System",
  "Local Network",
  "User assisted",
  "Passive",
  "Remote"
]
```

### Get Enumeration of Problem Product Security Exploitability

Client request:

```
GET /enumerations/problems/product-security/exploitabilities
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "Theoretical",
  "Conceptual",
  "Functional",
  "Automatic",
  "Worm"
]
```

### Get Enumeration of Problem Product Security Authentication



Client request:

```
GET /enumerations/problems/product-security/authentications
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
  "Authenticated",
  "Anonymous"
]
```

### Get Enumeration of Problem Product Security Impact

Client request:

```
GET /enumerations/problems/product-security/impacts
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
  "Denial of Service",
  "Minor Compromise",
  "Major Compromise",
  "Total Compromise"
]
```

### Get Enumeration of Problem Product Security Assets

Client request:

```
GET /enumerations/problems/product-security/assets
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=UTF-8
[
  "General",
  "Application",
  "Credentials",
  "User Information",
]
```





```
"System"  
]
```

### Get Enumeration of Problem Product Security UserBase

Client request:

```
GET /enumerations/problems/product-security/user-bases  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
X-API-Version: 1.0  
Content-Type: application/json; charset=UTF-8  
[  
  "Few Systems",  
  "Some Systems",  
  "Many Systems",  
  "Key Systems",  
  "Most Systems"  
]
```

### Get Enumeration of Problem Product Security Complexity

Client request:

```
GET /enumerations/problems/product-security/complexities  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
X-API-Version: 1.0  
Content-Type: application/json; charset=UTF-8  
[  
  "Complex",  
  "Straightforward",  
  "Simple"  
]
```

### Get Enumeration of Problem Product Security Visibility

Client request:

```
GET /enumerations/problems/product-security/visibilities  
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200
```



```
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "Internal",
  "External",
  "Public",
  "Exploit Published"
]
```

### Get Enumeration of Component Person Role

Client request:

```
GET /enumerations/components/person-roles
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "Owner",
  "EPM",
  "Administrator",
  "Screener",
  "Integrator",
  "Builder",
  "Verifier"
]
```

### Get Enumeration of Problem Privileges

Client request:

```
GET /enumerations/security/privileges
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "Super User",
  "Owner",
  "Administrator",
  "Reassignable (Full)",
  "Reassignable (Tree-limited)",
  "Assignable",
  "Read-only",
  "No Privilege"
]
```



]

### Get Enumeration of Group Types

Client request:

```
GET /enumerations/group-types
X-API-Version: 1.0
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json;charset=UTF-8
[
  "Access Group",
  "Work Group"
]
```

### Get Enumeration of Group Privilege

Client request:

```
GET /enumerations/groups/privileges
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json;charset=UTF-8
[
  "Assignable",
  "Reassignable(Tree-limited)",
  "Reassignable(Full)"
]
```

### Get Enumeration of Group Members

Client request:

```
GET /enumerations/groups/member-types
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json;charset=UTF-8
[
  "Person",
  "Work Group",
  "Access Group"
]
```



## Get Enumeration of Find Problem OrderBy Fields

Client request:

```
GET /enumerations/order-by/fields
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json;charset=UTF-8
[
  "assignee",
  "assigneeLastModifiedAt",
  "mustBeFixedInBuild",
  "fixedInBuild",
  "foundInBuild",
  "verifiedInBuild",
  "ccCount",
  "classification",
  "closedAt",
  "componentName",
  "componentVersion",
  "crashCount",
  "createdAt",
  "dri",
  "effortCurrentTotalEstimate",
  "effortExpended",
  "effortOriginalTotalEstimate",
  "effortPercentComplete",
  "effortRemaining",
  "epm",
  "event",
  "eventEndDate",
  "fixOrder",
  "hasThirdPartyImpact",
  "hasConfidentialContentImpact",
  "hasHumanInterfaceImpact",
  "hasLocalizationImpact",
  "hasImportExportImpact",
  "hasPatentReviewImpact",
  "hasNewAPIImpact",
  "hasNewSPIImpact",
  "hasOpenSourceImpact",
  "inActive",
  "keywords",
  "label",
  "lastModifiedAt",
  "milestone",
  "milestoneEndDate",
  "originalProblemID",
  "originator",
  "priority",
```



```

    "id",
    "isProblem",
    "proxy",
    "isReadByAssignee",
    "isReadByProxy",
    "blockedByCount",
    "blockingProblemCount",
    "duplicateProblemCount",
    "totalRelation",
    "reproducible",
    "resolution",
    "resolvedBy",
    "resolvedDate",
    "isApproved",
    "isAutoCalculated",
    "dateNeededCurrent",
    "dateNeededOriginal",
    "targetCompletionCurrent",
    "targetCompletionOriginal",
    "targetStartDate",
    "scheduleTestCase",
    "isUmbrella",
    "scheduleTestIDs",
    "assets",
    "attackVector",
    "authentication",
    "color",
    "complexity",
    "cwePrimary",
    "cweSecondary",
    "foundAt",
    "disclosedAt",
    "securityDRI",
    "exploitability",
    "externals",
    "impact",
    "isPrivacyIssue",
    "securityMasterID",
    "type",
    "userBase",
    "securityVerifier",
    "visibility",
    "buildInfo",
    "state",
    "substate",
    "targetMilestonePlaceholders",
    "targetMilestoneTotal",
    "taskOrder",
    "isThirdPartyAppRelated",
    "title",
    "isVerifiedByTester"
  ]

```

#### Get Enumeration of Recent Assignee Types



Client request:

```
GET /enumerations/problems/assignees/types
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=UTF-8
[
  "Previous Assignee 1",
  "Previous Assignee 2",
  "Previous Assignee 3",
  "Originator",
  "DRI",
  "Screener",
  "Integrator",
  "Builder",
  "Verifier"
]
```

#### Get Enumeration of Problem Protection Mask type

Client request:

```
GET /enumerations/problems/mask/types
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=UTF-8
[
  "read-only",
  "modifiable"
]
```

#### Get Enumeration of Recent Queries Types

Client request:

```
GET /enumerations/recent-queries/types
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=UTF-8
[
  "Problem",
  "Shared Report",
  "TSTT"
]
```

#### Get Enumeration of Shared Report Subscriber Types



Client request:

```
GET /enumerations/shared-reports/subscribers/types
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=UTF-8
[
  "Person",
  "Work Group",
  "Access Group",
  "Everyone"
]
```

### Get Enumeration of Shared Report Subscriber Permissions

Client request:

```
GET /enumerations/shared-reports/subscribers/permission
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=UTF-8
[
  "Read Only",
  "Read & Write",
  "Owner (Read & Write)",
  "No Access"
]
```

### Get Enumeration of Problems Assignee Types

Client request:

```
GET /enumerations/problems/assignees/types
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=UTF-8
[
  "Previous Assignee 1",
  "Previous Assignee 2",
  "Previous Assignee 3",
  "Originator",
]
```



```
"DRI",  
"Screener",  
"Integrator",  
"Builder",  
"Verifier"  
]
```

#### Get Enumeration of Recent Queries Executed As

Client request:

```
GET /enumerations/recent-queries/executed-as
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
Content-Type: application/json; charset=UTF-8
```

```
[  
  "Ad-Hoc",  
  "Saved"  
]
```





## 11. LABELS COLLECTION

### 11.1 Get Labels

#### [A] Description

This API provides a method to retrieve the list of labels which are in the active label set of the logged-in user. If there is no label in the active label set then an empty array will be returned.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

GET /labels

#### [D] Response Attributes

Response will be an array of label object. Each object will contain the below mentioned attributes.

Key	Description	Data Type
name	Name of the Label	String
id	ID of the Label	Integer
order	Order of the Label	Integer
color	Color of the Label. See <a href="#">Color Object</a> table for description.	Color Object

#### Color Object:

Color object must contains all the below attributes.

Key	Description	Data Type
red	Value of red color	Float
green	Value of green color	Float
blue	Value of blue color	Float
alpha	Value of alpha	Float

#### [E] Example

Client request:

GET /labels



X-API-Version: 1.3

Server response:

HTTP/1.1 200 OK

Status: 200

Content-Type: application/json; charset=utf-8

```
[
  {
    "id": 1084570,
    "order": 1,
    "color": {
      "red": 0.885322,
      "blue": 0.426327,
      "green": 0.117828,
      "alpha": 1
    },
    "name": "Red"
  },
  {
    "id": 1084571,
    "order": 2,
    "color": {
      "red": 0.094686,
      "blue": 0.834434,
      "green": 0.221608,
      "alpha": 1
    },
    "name": "Blue"
  },
  {
    "id": 1084572,
    "order": 3,
    "color": {
      "red": 0.063471,
      "blue": 0.210667,
      "green": 0.662984,
      "alpha": 1
    },
    "name": "Green"
  }
]
```

## 11.2 Label Set Labels

This set of APIs defines the manipulation of labels in the label set. It support both Shared Label Set as well as Personal Label Set. To manipulate labels in Personal Label Set, then the label set name should be 'My Personal Labels'.



### 11.2.1 Get Labels from Label Set

#### [A] Description

This API provides a method to retrieve the list of all labels which are in the specified label set. If there is no label in the label set then an empty array will be returned.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

```
GET /labels/<label-set-name>
```

#### [D] Response Attributes

Response attributes are same as described in the [Response Attributes](#) of section 11.1 Get Labels.

#### [E] Example

Client request:

```
GET /labels/LabelSet
X-API-Version: 1.3
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "id": 1084570,
    "order": 1,
    "color": {
      "red": 0.885322,
      "blue": 0.426327,
      "green": 0.117828,
      "alpha": 1
    },
    "name": "Red"
  },
  {
    "id": 1084571,
    "order": 2,
    "color": {
      "red": 0.094686,
      "blue": 0.834434,
      "green": 0.221608,
      "alpha": 1
    },
    "name": "Blue"
  }
]
```



```
    },  
    {  
      "id": 1084572,  
      "order": 3,  
      "color": {  
        "red": 0.063471,  
        "blue": 0.210667,  
        "green": 0.662984,  
        "alpha": 1  
      },  
      "name": "Green"  
    }  
  ]  
}
```

### 11.2.2 Add Label to Label Set

#### [A] Description

This API provides a method to add a label into specified label set. On success, the server responds with 201 Created and no response body.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

POST /labels/<label-set-name>

#### [D] Request Attributes

Key	Description	Data Type
name	Name of the Label	String
order	Order of the Label	Integer
color	Color of the Label. See <a href="#">Color Object</a> table for description.	Color Object

#### [E] Example

Client request:

```
POST /labels/LabelSet  
X-API-Version: 1.3  
Content-Type: application/json; charset=UTF-8  
{  
  "order": 1,  
  "color": {
```



```
        "red": 0.173284,  
        "blue": 0.055125,  
        "green": 0.407204,  
        "alpha": 1.000000  
    },  
    "name": "Label-1"  
}
```

Server response:

```
HTTP/1.1 201 Created  
Status: 201  
X-API-Version: 1.3
```

### 11.2.3 Edit Label in Label Set

#### [A] Description

This API provides a method to edit a label in the specified label set. On success, the server responds with 201 Created and no response body.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

```
PUT /labels/<label-set-name>/<label-name>
```

#### [D] Request Attributes

Request attributes are same as described in [Request Attributes](#) of section 11.2.2 Add Label to Label Set.

#### [E] Example

Client request:

```
PUT /labels/LabelSet/Label-1  
X-API-Version: 1.3  
Content-Type: application/json; charset=UTF-8  
{  
    "order": 1,  
    "color": {  
        "red": 0.373284,  
        "blue": 0.665125,  
        "green": 0.507204,  
        "alpha": 1.000000  
    },  
    "name": "Label"  
}
```

Server response:

```
HTTP/1.1 201 Created
```



Status: 201  
X-API-Version: 1.3

### 11.2.4 Delete Label from Label Set

#### [A] Description

This API provides a method to delete a label from specified label set. The request consists solely of a DELETE and URL path, without any request body. On success, the server responds with 204 No Content and no response body.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

```
DELETE /labels/<label-set-name>/<label-name>
```

#### [D] Examples

Client request:

```
DELETE /labels/LabelSet/Label  
X-API-Version: 1.3
```

Server response:

```
HTTP/1.1 204 No Content  
Status: 204  
X-API-Version: 1.3
```

## 11.3 Find Label Sets

#### [A] Description

This API is used to find label sets via a JSON POST interface. The client can specify a JSON hash that contains the fields against which to perform a search.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

```
POST /label-sets/find
```

#### [D] Request Attributes

The following problem attributes can be searched:



Key	Description	Data Type
name	Name of the Label Set	String
component	Component of the Label Set. This object can contains name, version and includeSubcomponents attributes.	Component Object
description	Description of the Label Set	String

#### [E] Response Attributes

Response will be an array of label set object. Each object will contain the below mentioned attributes.

Key	Description	Data Type
name	Name of the Label Set	String
id	ID of the Label Set	Integer
component	Component of the Label Set, object will contain name and version attributes.	Component Object
description	Description of the Label Set	String

#### [F] Examples

Client request:

```
POST /label-sets/find
Content-Type: application/json;charset=UTF-8
{
  "name": "Radar",
  "component": {
    "name": "Radar",
    "version": "Maintenance"
  },
  "description": "Radar"
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.3

[
  {
    "id": 76009,
    "component": {
      "name": "Radar",
      "version": "Maintenance"
    },
    "description": "Radar Maintenance Shared Label Set",
```



```

    "name": "Radar Maintenance"
  }
]

```

## 11.4 Subscribe Label Set

This section has APIs to get, subscribe and unsubscribe the label set to the current logged-in user.

### 11.4.1 Get Subscribed Label Sets

#### [A] Description

This API provides a method to get the list of subscribed label set of the current logged-in user. The request consists solely of a GET and URL path, without any request body. On success, the server responds with 200 OK and response body.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

```
GET /label-sets/subscribe
```

#### [E] Response Attributes

Response will be an array of label set object. Each object will contain the below mentioned attributes.

Key	Description	Data Type
name	Name of the Label Set	String
id	ID of the Label Set	Integer
component	Component of the Label Set, object will contain name and version attributes.	Component Object
description	Description of the Label Set	String
isEnabled	Is this label set enabled?	Boolean

#### [F] Examples

Client request:

```
GET /label-sets/subscribe
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
```





X-API-Version: 1.3

```
[
  {
    "id": "235575",
    "component": {
      "name": "Auto iQA Root Component",
      "version": "7.3"
    },
    "description": "Label Set For Automation Purpose",
    "name": "AutoiQALabelSet",
    "isEnabled": true
  },
  {
    "id": "235572",
    "component": {
      "name": "Radar 7.3 testing",
      "version": "7.3"
    },
    "description": "asdasdasdasd",
    "name": "dasdadasd",
    "isEnabled": true
  },
  {
    "id": "235571",
    "component": {
      "name": "abcd",
      "version": "7.3"
    },
    "description": "aaaaaaa",
    "name": "No Labels",
    "isEnabled": false
  }
]
```

## 11.4.2 Subscribe a Label Set

### [A] Description

This API provides a method to subscribe a label set to the current logged-in user. The request consists solely of a POST and URL path, without any request body. On success, the server responds with 201 Created and no response body. Error message will be thrown if trying to subscribe 'My Personal Labels' label set.

### [B] Schedule

Required for version 1.3

### [C] URL Scheme

POST /label-sets/<label-set-name>/subscribe

**[D] Examples**

Client request:

```
POST /label-sets/Radar%20Maintenance/subscribe
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.3
```

**11.4.3 Unsubscribe a Label Set****[A] Description**

This API provides a method to unsubscribe a label set to the current logged-in user. The request consists solely of a DELETE and URL path, without any request body. On success, the server responds with 204 No Content and no response body.

**[B] Schedule**

Required for version 1.3

**[C] URL Scheme**

```
DELETE /label-sets/<label-set-name>/unsubscribe
```

**[D] Examples**

Client request:

```
DELETE /label-sets/Radar%20Maintenance/subscribe
```

Server response:

```
HTTP/1.1 204 No Content
Status: 204
X-API-Version: 1.3
```

**11.5 Active Label Set**

This section has APIs to get and set active label set to the current logged-in user.

**11.5.1 Get Active Label Set****[A] Description**

This API provides a method to get the details of the active label set of the current logged-in user. The request consists solely of a GET and URL path, without any request body. On success, the server responds with 200 OK and response body.

**[B] Schedule**



Required for version 1.3

#### [C] URL Scheme

GET /label-sets/active

#### [D] Response Attributes

Response will be an object which will contain the below mentioned attributes.

Key	Description	Data Type
name	Name of the Label Set	String
id	ID of the Label Set	Integer
component	Component of the Label Set, object will contain name and version attributes.	Component Object
description	Description of the Label Set	String
isEnabled	Is this label set enabled?	Boolean
labels	Labels of the Label Set. Attributes are same as described in the <a href="#">Response Attributes</a> of section 11.1 Get Labels	Array of Label Object

#### [E] Examples

Client request:

GET /label-sets/active

Server response:

HTTP/1.1 200 OK

Status: 200

X-API-Version: 1.3

```
{
  "id": 235575,
  "component": {
    "name": "Auto iQA Root Component",
    "version": "7.3"
  },
  "description": "Label Set For Automation Purpose",
  "labels": [
    {
      "id": 1274976,
      "order": 1,
      "color": {
        "red": 0.710407,
        "blue": 0.114016,
        "green": 0.114016,
        "alpha": 1
      }
    }
  ]
}
```



```

    },
    "name": "AutoiQALabelSet"
  },
  {
    "id": 1275718,
    "order": 2,
    "color": {
      "red": 0.500873,
      "blue": 0.773756,
      "green": 0.08174,
      "alpha": 1
    },
    "name": "Test"
  }
],
"name": "AutoiQALabelSet",
"isEnabled": true
}

```

### 11.5.2 Set Active Label Set

#### [A] Description

This API provides a method to set a label set as a active label set to the current logged-in user. The request consists solely of a POST and URL path, without any request body. On success, the server responds with 201 Created and no response body.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

```
POST /label-sets/<label-set-name>/active
```

#### [D] Examples

Client request:

```
POST /label-sets/Radar%20Maintenance/active
```

Server response:

```

HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.3

```



## 12. FAVORITE PREFERENCE APIS

### 12.1 Get Favourite Shared Report

#### [A] Description

This API provides a method to retrieve the list of favourite shared report for logged-in user. If there is no favourite shared report then an empty array will be returned.

A new attribute 'type' added in response attribute of Get Favourite Shared Report API. It will contain any one of report type like 'Problem Query', 'AppleScript', 'SQL' or 'Folder'. If Favourite Shared Report contains a folder then it will contain type as 'Folder'.

To fetch specific report in response based on reportType, a new header 'X-Report-Type' is added. 'X-Report-Type' can have any one of 'Problem Query', 'AppleScript', 'SQL' 'Folder'. If 'X-Report-Type' contains any other value then an error message will be shown.

If 'X-Report-Type' is not passed in request then response will contain all reports with folders added as users favourite shared report. Folders will not be filtered in response if 'X-Report-Type' is used to maintain folder structure used by User.

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

GET /favorites/shared-reports

#### [D] Response Attributes

Response will be an array of shared-report object. Each object will contain the below mentioned attributes.

Key	Description	Data Type
id	ID of favorite shared report	Integer
name	Name of favorite shared report	String
owner	Owner of shared report	Person Object
isFolder	Is object a folder or report	Boolean
folders	Array of shared report object. It will be added only if isFolder attribute is true.	Array of object
type	Type of shared report. Will contain any one of 'Problem Query', 'AppleScript', 'SQL', 'Folder'	String

### [E] Example

Client request:

```
GET /favorites/shared-reports
X-API-Version: 1.3
```

Server response:

```

HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[
  {
    "id": 112213,
    "name": "TSTT Reports",
    "owner": {
      "lastName": "Tester1",
      "email": "radartester01@gmail.com",
      "type": "Contractor",
      "firstName": "Radar",
      "dsid": 1118580968
    },
    "isFolder": true,
    "folders": [
      {
        "id": 1414441,
        "name": "TSTT: Export Cases from Test Suite selection",
        "owner": {
          "lastName": "Tester1",
          "email": "radartester01@gmail.com",
          "type": "Contractor",
          "firstName": "Radar",
          "dsid": 1118580968
        }
      }
    ],
    {
      "id": 235255,
      "name": "TSTT: Export Cases from scheduledTest selection",
      "owner": {
        "lastName": "Tester1",
        "email": "radartester01@gmail.com",
        "type": "Contractor",
        "firstName": "Radar",
        "dsid": 1118580968
      }
    }
  ]
},
{
  "id": 10995,
  "name": "Untitled",
  "owner": {
    "lastName": "Tester1",
    "email": "radartester01@gmail.com",
    "type": "Contractor",
    "firstName": "Radar",
    "dsid": 1118580968
  }
}
]

```



GET /favorites/shared-reports  
 X-API-Version: 1.4  
 X-Report-Type: Problem Query

Server response:

HTTP/1.1 200 OK  
 Status: 200  
 X-API-Version: 1.4  
 Content-Type: application/json; charset=utf-8

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8

[{"id": 112213,
  "name": "TSTT Reports",
  "owner": {
    "lastName": "Tester1",
    "email": "radartester01@gmail.com",
    "type": "Contractor",
    "firstName": "Radar",
    "dsid": 1118580968
  },
  "isFolder": true,
  "type": "Folder",
  "folders": [{
    "id": 1414441,
    "name": "TSTT: Export Cases from Test Suite selection",
    "owner": {
      "lastName": "Tester1",
      "email": "radartester01@gmail.com",
      "type": "Contractor",
      "firstName": "Radar",
      "dsid": 1118580968
    },
    "type": "Problem Query"
  }],
  "type": "Problem Query"
},
{
  "id": 235255,
  "name": "TSTT: Export Cases from scheduledTest selection",
  "owner": {
    "lastName": "Tester1",
    "email": "radartester01@gmail.com",
    "type": "Contractor",
    "firstName": "Radar",
    "dsid": 1118580968
  },
  "type": "Problem Query"
}]
{
  "id": 10995,
  "name": "test",
```



```
    "owner": {
      "lastName": "Tester1",
      "email": "radartester01@gmail.com",
      "type": "Contractor",
      "firstName": "Radar",
      "dsid": 1118580968
    },
    "type": "Problem Query"
  ]
```

## 12.2 Add Favourite For Shared Reports

### [A] Description

This API provides a method to add shared reports to favourites of logged-in user.

### [B] Schedule

Required for version 1.4

### [C] URL Scheme

PUT /favorites/shared-reports/{report\_id}

### [D] Example

Client request:

```
PUT /favorites/shared-reports/12345
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
X-API-Version: 1.4
```

## 12.3 Remove Favourite For Shared Reports

### [A] Description

This API provides a method to remove shared reports from favourites of logged-in user.

### [B] Schedule

Required for version 1.4

### [C] URL Scheme

DELETE /favorites/shared-reports/{report\_id}



**[D] Example**

Client request:

```
DELETE /favorites/shared-reports/12345
```

Server response:

```
HTTP/1.1 204 No Content
Server: Apache-Coyote/1.1
X-API-Version: 1.4
```

**12.4 Get Favourite People****[A] Description**

This API provides a method to retrieve the list of favourite people added for logged-in user. If there is no favourite people then an empty array will be returned.

**[B] Schedule**

Required for version 1.2

**[C] URL Scheme**

```
GET /favorites/people
```

**[D] Response Attributes**

Response will be an array of person object. Each object will contain the below mentioned attributes.

Key	Description	Data Type
dsid	DSID of the person	Integer
firstName	First name of the person	String
lastName	Last name of the person	String
email	Email-Id of the person	String
type	Type of person	String

**[E] Example**

Client request:

```
GET /favorites/people
X-API-Version: 1.2
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
```



```
Content-Type: application/json; charset=utf-8
[{
  "lastName": "Tester1",
  "email": "radartester01@gmail.com",
  "type": "Contractor",
  "firstName": "Radar",
  "dsid": 1118580968
},
{
  "lastName": "Tester2",
  "email": "radartester02@gmail.com",
  "type": "Contractor",
  "firstName": "Radar",
  "dsid": 1118580920
}]
```

## 12.5 Add To Favourite For People

### [A] Description

This API provides a method to add a person to the list of favourite people for logged-in user.

### [B] Schedule

Required for version 1.4

### [C] URL Scheme

```
PUT /favorites/people/<dsid>
```

### [D] Example

Client request:

```
PUT /favorites/people/102100455
```

Server response:

```
HTTP/1.1 201 Created
```

## 12.6 Remove From Favourite For People

### [A] Description

This API provides a method to remove a person from the list of favourite people for logged-in user.

### [B] Schedule

Required for version 1.4

**[C] URL Scheme**

```
DELETE /favorites/people/{dsid}
```

**[D] Example**

Client request:

```
DELETE /favorites/people/1195039228
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 204 No Content
X-API-Version: 1.4
```

**12.7 Get Favourite Component****[A] Description**

This API provides a method to retrieve the list of favourite components added for logged-in user. If there is no favourite component added then an empty array will be returned.

**[B] Schedule**

Required for version 1.2

**[C] URL Scheme**

```
GET /favorites/components
```

**[D] Response Attributes**

Response will be an array of component object. Each object will contain the below mentioned attributes.

Key	Description	Data Type
id	ID of component	Integer
name	Name of component	String
version	Version of component	String
isClosed	Is Component closed	Boolean
isDefault	Is Component default	Boolean

**[E] Example**



Client request:

```
GET /favorites/components
X-API-Version: 1.2
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[ {
  "id": 324234,
  "name": "Radar",
  "version": "7.4",
  "isClosed": false,
  "isDefault": true
},
{
  "id": 342345,
  "name": "Radar",
  "version": "7.3",
  "isClosed": false,
  "isDefault": false
} ]
```

## 12.8 Add To Favourite For Components

### [A] Description

This API provides a method to add component to the list of favourite components for logged-in user.

### [B] Schedule

Required for version 1.4

### [C] URL Scheme

```
PUT /favorites/components/{Component-Name}/{Component-Version}
```

### [D] Example

Client request:

```
PUT /favorites/components/Radar/Automation
```

Server response:

```
HTTP/1.1 201 Created
```

## 12.9 Remove From Favourite For Components

### [A] Description



This API provides a method to remove the component from the list of favorite components for logged-in user.

**[B] Schedule**

Required for version 1.4

**[C] URL Scheme**

```
DELETE /favorites/components/{componentName}/{componentVersion}
```

**[E] Example**

Client request:

```
DELETE /favorites/components/YComponent/Y
X-API-Version: 1.4
```

Server response:

```
HTTP/1.1 204 No Content
X-API-Version: 1.4
```

## 12.10 Get Favourite Other related Items

**[A] Description**

This API provides a method to retrieve the list of favourite other related items added for logged-in user. If there is no favourite other related item added then an empty array will be returned.

**[B] Schedule**

Required for version 1.2

**[C] URL Scheme**

```
GET /favorites/other-related-systems
```

**[D] Response Attributes**

Response will be an array of label object. Each object will contain the below mentioned attributes.

Key	Description	Data Type
system	System name of other related item	String
urlScheme	URL scheme of other related item	String
isGlobal	Is other related item global	Boolean

**[E] Example**

Client request:

```
GET /favorites/other-related-systems
X-API-Version: 1.2
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json; charset=utf-8
[ {
  "system": "Radar",
  "urlScheme": "<rdar://problem/<ID>>>",
  "isGlobal": false
},
{
  "system": "Espresso",
  "urlScheme": "<EXP://ticket/<ID>>>",
  "isGlobal": true
}]
```

**12.11 Add Favourite For Other related Items****[A] Description**

This API provides a method to add other related item to favourites for logged-in user.

**[B] Schedule**

Required for version 1.4

**[C] URL Scheme**

```
PUT /favorites/other-related-systems/<name>/<urlscheme>
```

**[D] Example**

Client request:

```
PUT /favorites/other-related-systems/Radartip/radar.com
```

Server response:

```
HTTP/1.1 201 Created
```

**12.12 Remove Favourite For Other related Items****[A] Description**

This API provides a method to remove other related item from favorites of logged-in user.

**[B] Schedule**

Required for version 1.4

**[C] URL Scheme**

```
DELETE /favorites/other-related-systems/{name}/{URLScheme}
```

**[D] Example**

Client request:

```
DELETE /favorites/other-related-systems/Radartip/radar.com
```

Server response:

```
HTTP/1.1 204 No Content  
Server: Apache-Coyote/1.1  
X-API-Version: 1.4
```

**12.13 Get Favourite Layouts****[A] Description**

This API provides a method to retrieve the list of Favorite layouts for logged-in user. If there is no Favorite layouts then an empty array will be returned.

**[B] Schedule**

Required for version 1.5

**[C] URL Scheme**

```
GET /favorites/layouts
```

**[D] Response Attributes**

Response will be an array of layout object. Each object will contain the below mentioned attributes.

Key	Description	Data Type
id	ID of favorite layout	Integer
name	Name of favorite layout	String
owner	Owner of layout	Person Object
isFolder	Is object a folder or layout	Boolean



Key	Description	Data Type
layouts	Array of layout object. It will be added only if isFolder attribute is true.	Array of layout object

### [E] Example

Client request:

GET /favorites/layouts

Server response:

HTTP/1.1 200 OK

Status: 200

Content-Type: application/json; charset=utf-8

```
[{
  "id": 112213,
  "name": "New Folder",
  "owner": {
    "lastName": "Tester1",
    "email": "radartester01@gmail.com",
    "type": "Contractor",
    "firstName": "Radar",
    "dsid": 1118580968
  },
  "isFolder": true,
  "layouts": [{
    "id": 1414441,
    "name": "ADC Layout",
    "owner": {
      "lastName": "Tester1",
      "email": "radartester01@gmail.com",
      "type": "Contractor",
      "firstName": "Radar",
      "dsid": 1118580968
    }
  ]
},
{
  "id": 235255,
  "name": "ADC Layout111",
  "owner": {
    "lastName": "Tester1",
    "email": "radartester01@gmail.com",
    "type": "Contractor",
    "firstName": "Radar",
    "dsid": 1118580968
  }
}]
{
  "id": 10995,
  "name": "ADC Layout123",
  "owner": {
    "lastName": "Tester1",
```





```
"email": "radartester01@gmail.com",  
"type": "Contractor",  
"firstName": "Radar",  
"dsid": 1118580968  
}]
```

## 12.14 Add Layouts to Favorites

### [A] Description

This API provides a method to add layout to Favorites of logged-in user. As this feature is only used by clients and not in automation tasks, it will be considered Optimized for Radar Clients.

### [B] Schedule

Required for version 1.5

### [C] URL Scheme

PUT /favorites/layouts/<layout-ID>

### [D] Example

Client request:

```
PUT /favorites/layouts/12345
```

Server response:

```
HTTP/1.1 201 Created  
Status: 201
```

## 12.15 Remove Favorite For Layouts

### [A] Description

This API provides a method to remove layout from Favorites of logged-in user. This API will only delete layouts and folders will not be deleted.

### [B] Schedule

Required for version 1.5

### [C] URL Scheme

DELETE /favorites/layouts/<layout-ID>

### [D] Example

Client request:



DELETE /favorites/layouts/12345

Server response:

HTTP/1.1 204 No Content



## 13. GROUP

### 13.1 Get Work-Group Details

#### [A] Description

This API provides a method to fetch details of the work-group based on the Group name. The response will contain only default attributes mentioned in response parameter section. To get the non-default parameter `X-Fields-Requested` header need to be used with the required parameter.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

GET /groups/work-groups/<group-name>

#### [D] Response Parameters

Parameter	Description	Data Type	Default
name	Name of the Group	String	Y
description	Description of the Group	String	Y
assignToRole Holder	Is Assign to role holder for current state when component changes?	Boolean	N
resetStateOn Component Change	Is Reset state to Analyse on component change?	Boolean	N
autoAssignTo Originator On Verify	Is Auto-Assign to Originator on Verify?	Boolean	N
roles	An object containing all the roles associated with the group	Roles Object	Y
administrators	Refer table 1.3 for Admin/Members Object	JSON Array	N
members	Refer table 1.3 for Admin/Members Object	JSON Array	N
components	Refer table 1.4 for Component Object	JSON Array	N

**Table 1.1 Roles Object**



Key	Description	Data Type
owner	Owner of the Group	Person Object
screener	Screener of Group.	Person Object
integrator	Integrator of Group.	Person Object
builder	Builder of Group.	Person Object
verifier	Verifier of Group.	Person Object

**Table 1.2 Person Object**

Key	Description	Data Type
lastName	Last Name of the person	Object
firstName	First Name of the person	String
type	type of access (Employee, No Access)	String
email	Email Address of the person	String
dsid	dsId of the person	String

**Table 1.3 Admin / Members Object**

Key	Description	Data Type
person	Refer table 1.1 for Person Object	Object
department	Department to which employee belongs to	String
lastModifiedAt	Last Modified Date	ISO 8601 date string
lastModifiedBy	Last Modified Person Object	Person Object
privilege	Privilege held by admin will be Administrator and for members it will be enumeration mentioned in 2.22.11 Get Group Privilege Enumerations	Enumerated Values

**Table 1.4 Component Object**

Key	Description	Data Type
name	The component name	String



Key	Description	Data Type
version	The component version	String
isClosed	Indicates whether the association is closed	Boolean
description	Description of the Association	String

## [F] Examples

### Client request with Default fields :

GET /groups/work-groups/Radar

Server response:

```
{
  "description": "Versions: 3, 4, 5, 6, 7, etc.",
  "roles": {
    "builder": {
      "lastName": "Ewalt",
      "email": "ewalt@apple.com",
      "dsid": 8794,
      "firstName": "Alan",
      "type": "Employee"
    },
    "owner": {
      "lastName": "Ewalt",
      "email": "ewalt@apple.com",
      "dsid": 8794,
      "firstName": "Alan",
      "type": "Employee"
    },
    "verifier": {
      "lastName": "Ewalt",
      "email": "ewalt@apple.com",
      "dsid": 8794,
      "firstName": "Alan",
      "type": "Employee"
    },
    "integrator": {
      "lastName": "Ewalt",
      "email": "ewalt@apple.com",
      "dsid": 8794,
      "firstName": "Alan",
      "type": "Employee"
    },
    "screener": {
      "lastName": "Ewalt",
      "email": "ewalt@apple.com",
      "dsid": 8794,
      "firstName": "Alan",
      "type": "Employee"
    }
  }
}
```



```

    }
  },
  "name": "Radar"
}

```

**Client request with X-Fields-Requested :**

GET /groups/work-groups/Radar

X-Fields-Requested:name,administrators

```

{
  "name": "Radar",
  "administrators": [
    {
      "person": {
        "lastName": "Ewalt",
        "email": "ewalt@apple.com",
        "dsid": 8794,
        "firstName": "Alan",
        "type": "Employee"
      },
      "department": "R&D IS&T",
      "privilege": "Administrator",
      "lastModifiedBy": {
        "lastName": "Ewalt",
        "email": "ewalt@apple.com",
        "dsid": 8794,
        "firstName": "Alan",
        "type": "Employee"
      },
      "lastModifiedAt": "2013-03-07T21:06:00+0000"
    },
    {
      "person": {
        "lastName": "Iwata",
        "email": "xxxxxxx@apple.com",
        "dsid": 2648,
        "firstName": "Cyndi",
        "type": "Employee"
      },
      "department": "R&D IS&T",
      "privilege": "Administrator",
      "lastModifiedBy": {
        "lastName": "Ewalt",
        "email": "ewalt@apple.com",
        "dsid": 8794,
        "firstName": "Alan",
        "type": "Employee"
      },
      "lastModifiedAt": "2011-01-14T23:01:00+0000"
    }
  ]
}

```



## 13.2 Get Access-Group Details

### [A] Description

This API provides a method to fetch details of the access-group based on the group name. The response will contain only default attributes mentioned in response parameter section. To get the non-default parameter `X-Fields-Requested` header need to be used with the required parameter.

### [B] Schedule

Required for version 1.4

### [C] URL Scheme

GET /groups/access-groups/<group-name>

### [D] Response Parameters

Parameter	Description	Data Type	Default
name	Name of the Group	String	Y
description	Description of the Group	String	Y
syncWithDS	Is Sync with DS Group?	Boolean	N
roles	An object containing roles associated with the group	Roles Object	Y
administrators	Refer table 1.3 for Admin/Members Object	JSON Array	N
members	Refer table 1.3 for Admin/Members Object	JSON Array	N
components	Refer table 1.4 for Component Object	JSON Array	N
restrictedMilestones	Refer table 1.5 for Restricted Milestone Object.	JSON Array	N

**Table 1.1 Roles Object**

Key	Description	Data Type
owner	Owner of the Group	Person Object

**Table 1.2 Person Object**



Key	Description	Data Type
lastName	Last Name of the person	Object
firstName	First Name of the person	String
type	type of access (Employee, No Access)	String
email	Email Address of the person	String
dsid	dsId of the person	String

**Table 1.3 Admin / Members Object**

Key	Description	Data Type
person	Refer table 1.2 for Person Object	Object
department	Department to which employee belongs to	String
lastModifiedAt	Last Modified Date	ISO 8601 date string
lastModifiedBy	Last Modified Person Object	Person Object

**Table 1.4 Component Object**

Key	Description	Data Type
name	The component name	String
version	The component version	String
isClosed	Indicates whether the association is closed	Boolean
privilege	Privilege held by component. Enumeration are mentioned in Get Milestone Privilege Enumerations	Enumerated Values
description	Description of the Association	String

**Table 1.5 Restricted Milestone Object**

Key	Description	Data Type
name	Name of the milestone	String
component	A component object	Component Object





Key	Description	Data Type
beginsAt	Milestones Begin Date	ISO 8601 date string
endsAt	Milestones Begin Date	ISO 8601 date string
isClosed	Is Closed?	Boolean
privilege	privilege held by restricted milestone. Enumeration are mentioned in Get Milestone Privilege Enumerations	Enumerated Values

## [F] Examples

### Client request with Default fields :

```
GET /groups/access-groups/Radar%2010g
```

Server response:

```
{
  "description": "Access group for member access to Radar.",
  "roles": {
    "owner": {
      "lastName": "Ewalt",
      "email": "ewalt@apple.com",
      "dsid": 8794,
      "firstName": "Alan",
      "type": "Employee"
    }
  },
  "name": "Radar 10g"
}
```

### Client request with X-Fields-Requested :

```
GET /groups/access-groups/Radar%2010g
```

```
X-Fields-Requested:name,members
```

```
{
  "name": "Radar 10g",
  "members": [
    {
      "person": {
        "lastName": "Butah",
        "email": "xxxxxxx@apple.com",
        "dsid": 142917308,
        "firstName": "Jon",
        "type": "No Access"
      },
      "department": "Developer Publications",
      "lastModifiedBy": {
```



```

        "lastName": "Butah",
        "email": "xxxxxxx@apple.com",
        "dsid": 142917308,
        "firstName": "Jon",
        "type": "No Access"
    },
    "lastModifiedAt": "2010-05-11T17:50:00+0000"
}
]
}

```

### 13.3 Create Work-Group

#### [A] Description

This API provides a method to create new work-group. The fields mentioned in request parameters section are supported for creating new work-group. All fields mentioned as mandatory should be passed in request otherwise error message will be shown.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

POST /groups/work-groups

#### [D] Request Parameters

Request Object will contain the below mentioned fields. All the Mandatory fields should be present in request object.

Parameter	Description	Data Type	Mandatory
name	Name of the Group	String	Y
description	Description of the Group	String	Y
assignToRole Holder	Is Assign to role holder for current state when component changes?	Boolean	N
resetStateOn Component Change	Is Reset state to Analyse on component change?	Boolean	N
autoAssignTo Originator On Verify	Is Auto-Assign to Originator on Verify?	Boolean	N



roles	An object containing all the roles associated with the group	Roles Object	N
-------	--	--------------	---

### Roles Object

Key	Description	Data Type	Mandatory
owner	Owner of the Group	Integer	Y
screener	Screener of Group.	Integer	N
integrator	Integrator of Group.	Integer	N
builder	Builder of Group.	Integer	N
verifier	Verifier of Group.	Integer	N

### [E] Examples

#### For Creating Work Group

Client request:

```
POST /groups/work-groups
{
  "name": "Test 2666_4",
  "description": "Creation of Work Group through rest API",
  "assignToRoleHolder":true,
  "resetStateOnComponentChange":true,
  "autoAssignToOriginatorOnVerify":false,
  "roles": {
    "owner": 1118580968,
    "screener": 1118580968,
    "integrator": 1118580968,
    "builder": 1118580968,
    "verifier": 1118580968"
  }
}
```

Server response:

```
HTTP/1.1 201 Created
```

## 13.4 Create Access-Group

### [A] Description

This API provides a method to create new access-group. The fields mentioned in request parameters section are supported for creating new access-group. All fields mentioned as mandatory should be passed in request otherwise error message will be shown.

**[B] Schedule**

Required for version 1.4

**[C] URL Scheme**

POST /groups/access-groups

**[D] Request Parameters**

Request Object will contain the below mentioned fields. All the Mandatory fields should be present in request object.

Parameter	Description	Data Type	Mandatory
name	Group Name	String	Y
description	Group Description	String	Y
dsGroup	Name of the DS group to synchronize this group. When 'dsGroup' is passed in the request then 'name' and 'description' fields should not be passed in the request.	String	N

**[E] Examples****For Creating Access Group**

Client request:

```
POST /groups/access-groups
{
  "name": "Test 2701_3",
  "description": "Creation of Access Group through rest API",
}
```

Server response:

HTTP/1.1 201 Created

**For Creating Access Group with invalid DS group**

```
POST /groups/access-groups
{
  "dsGroup": "test-group-that-doesnt-exist",
}
```

Server response:



```
HTTP/1.1 400 Bad Request
{
  "message": "The DS group provided doesn't exist",
  "title": "DS Group doesn't exist",
  "status": "400 Bad Request",
  "help": "View documentation at http://radar.apple.com/"
}
```

## 13.5 Update Work-Group

### [A] Description

This API provides a method to update work-group based on the request parameters.

### [B] Schedule

Required for version 1.4

### [C] URL Scheme

PUT /groups/work-groups/<group-name>

### [D] Request Parameters

Parameter	Description	Data Type
name	Group Name	String
description	Group Description	String
assignToRoleHolder	Assign to the role holder for current state when component changes	Boolean
resetStateOnComponentChange	Reset state to analyze on component change	Boolean
autoAssignToOriginatorOnVerify	Auto-Assign to Originator on verify	Boolean
roles	A Roles object	Roles Object

### Roles Object



Key	Description	Data Type
owner	Owner of the Group	Integer
screener	Screener of Group.	Integer
integrator	Integrator of Group.	Integer
builder	Builder of Group.	Integer
verifier	Verifier of Group.	Integer

### [E] Examples

#### For Updating Work Group

Client request:

```
PUT /groups/work-groups/Test2666_4
{
  "description": "Creation of Work Group through rest API",
  "isAssignToHolder":true,
  "isResetState":true,
  "isAutoAssign":false
}
```

Server response:

```
HTTP/1.1 201 Created
```

## 13.6 Update Access-Group

### [A] Description

This API provides a method to update access-group based on the request parameters.

### [B] Schedule

Required for version 1.4

### [C] URL Scheme

```
PUT /groups/access-groups/<group-name>
```

### [D] Request Parameters

Parameter	Description	Data Type
-----------	-------------	-----------



name	Group Name	String
description	Group Description	String
roles	A Roles object. Only owner can be passed in these role object.	Roles Object
dsGroup	Name of the DS group to synchronize this group. When 'dsGroup' is passed in the request then 'name' and 'description' and roles, fields should not be passed in the request.	String

### [E] Examples

#### For Updating Access Group

Client request:

```
PUT /groups/access-groups/Test2701_3
{
  "roles": {"owner":1118580968}
}
```

Server response:

```
HTTP/1.1 201 Created
```

## 13.7 Add Persons To Members

### [A] Description

This API provides a method to add person as a Member of Group. On success, the server responds with 201 Created and no response body.

### [B] URL Scheme

```
POST /groups/work-groups/<group-name>/members
POST /groups/access-groups/<group-name>/members
```

### [C] Request Attributes

The request consists of an array of objects with the following attributes:



Key	Description	Data Type
type	The type of the person or group to be added. Type can be "person", "Work Group" or "Access Group". Enumeration is mentioned in 2.22.10. Get Group Members Enumerations section	Enumerated Value
dsid	If type is "person", this will be the DSID of the person. If type is not "person", this will be ignored.	Integer
name	If type is "Work Group" or "Access Group", this will be the name of the group. If type is "person", this will be ignored.	String
privilege	If type is "person" privilege of the person any one of "Assignable", "Reassignable(Tree-limited)" or "Reassignable(Full)". This field can be passes only for Work Group(not for Access Group). Enumeration is mentioned in 2.22.11. Get Group Privilege Enumerations section	Enumerated Value

#### [D] Examples

##### Add Person to Member:

Client request:

```
POST /groups/work-groups/Test2/members
Content-Type: application/json;charset=UTF-8
[
  {
    "dsid": 8794,
    "type": "person",
    "privileges": "Assignable"
  },
  {
    "name": "RAM DAC All",
    "type": "Access Group"
  }
]
```

Server response:

```
HTTP/1.1 201 Created
Status: 201
```

Client request:

```
POST /groups/access-groups/Test2/members
Content-Type: application/json;charset=UTF-8
[
```





```
{
  "dsid": 8794,
  "type": "person",
  "name": "ds-group",
  "privileges": "Assignable"
}
```

Server response:

```
HTTP/1.1 400 Bad Request
Status: 400
{
  "message": "A DS group and a DSID cannot be passed at the same time.",
  "title": "Invalid fields.",
  "status": "400 Bad Request",
  "help": "View documentation at http://radar.apple.com/"
}
```

### 13.8 Set Privilege for the Member

#### [A] Description

This API provides a method to set privileges of the person added as Members of Group(only for Work Group). On success, the server responds with 200 Success and no response body.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

```
PUT /groups/work-groups/<group-name>/members/<dsid>
```

#### [D] Request Attributes

Key	Description	Data Type
privilege	privilege of the person any one of among "Assignable", "Reassignable(Tree-limited)" or "Reassignable(Full)". Enumeration is mentioned in 2.22.11. Get Group Privilege Enumerations section	Enumerated Value

#### [E] Example

Client request:

```
PUT /groups/work-groups/Test2/members/102100455
{
  "privileges": "Assignable"
}
```



Server response:

```
HTTP/1.1 200 Success
Status: 200
```

### 13.9 Remove Group Members

#### [A] Description

This API provides a method to remove person from Members of Group .On success, the server responds with 204 No Content and no response body.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

```
DELETE /groups/work-groups/<group-name>/members/{dsid}
DELETE /groups/access-groups/<group-name>/members/{dsid}
```

#### [D] Example

```
DELETE /work-groups/TestGroup/members/102100455
```

Server response:

```
HTTP/1.1 204 No Content
```

### 13.10 Add Administrator

#### [A] Description

This API provides a method to add a person to the Administrator of Group.On success, the server responds with 201 Created and no response body.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

```
PUT /groups/work-groups/<group-name>/administrators/<dsid>
PUT /groups/access-groups/<group-name>/administrators/<dsid>
```

#### [D] Example

Client request:

```
PUT /groups/work-groups/TestGroup/administrators/102100455
```



Server response:

```
HTTP/1.1 201 Created
```

### 13.11 Remove Administrator

#### [A] Description

This API provides a method to remove person from Administrator of group .On success, the server responds with 204 No Content and no response body.

#### [B] Schedule

Required for version 1.4

#### [C] URL Scheme

```
DELETE /groups/work-groups/<group-name>/administrators/{dsid}
DELETE /groups/access-groups/<group-name>/administrators/{dsid}
```

#### [D] Example

```
DELETE /groups/work-groups/TestGroup/administrators/857845689
```

Server response:

```
HTTP/1.1 204 No Content
Server: Apache-Coyote/1.1
```

### 13.12 Find Group

#### [A] Description

This API provides a method to search for Groups based on any request parameter provided. The response of this API contains the array of object satisfying request parameters.

Parameter 'type' can be either String or Object. It can support 4 operators ie. it can include any one of (gt, lt, eq, neq). By default the parameter 'type' will support 'eq' operator.

For Example-

```
{
  "name": "Product",
  "type": {
    "gt": "Access Group"
  }
}
```

OR



```
{
  "name": "Product",
  "type": "Access Group"
}
```

#### [B] Schedule

Required for version 1.3

#### [C] URL Scheme

POST /groups/find

#### [D] Request Parameters

The Request body will contain any of the below parameters.

Parameter	Description	Data Type
name	Name of the Group	String
type	Type of the Group. It can be either Access Group or Work Group. It can includes any one of (gt, lt, eq, neq)	String or Object
description	Description of Group	String
dsGroupName	DS Group Name	String
hasExternalUsers	Boolean value for external user	Boolean
isSyncDSGroup	Boolean value for DS Group	Boolean
personnel	Refer table 1.1 for personnel Object	Object or Integer
component	Refer table 1.2 for component Object	Object

**Table 1.1 Personnel Object**

Key	Description	Data Type
dsid	DSID of the person	Integer
personRole	Role of the person in Group. It can includes any one of (gt, lt, eq, neq)	String or Object

**Table 1.2 Component Object**

Key	Description	Data Type
name	The component's name.	String
version	The component's version.	String

This table lists the operators that apply to parameters

Data Type	Supported Operators
type	gt, lt, eq, neq
personRole	gt, lt, eq, neq

**[E] Response Parameters**

Parameter	Description	Data Type
id	ID of the Group	Integer
name	Name of the Group	String
type	Type of the Group	String
isActive	Is Group active?	Boolean
allowExternals	Is Group External?	Boolean
description	Description of the Group	String or null

**[F] Examples**

Client request:

```
POST /groups/find
X-API-Version: 1.3
Content-Type: application/json; charset=utf-8

{
  "name": "Radar",
  "type": {
    "eq": "Work Group"
  },
  "description": "Versions: 3, 4, 5, 6, 7, etc.",
  "dsGroupName": "DS Group",
  "hasExternalUsers": true,
```



```
"isSyncDSGroup": true,
"personnel": {
  "dsid": 1118581234,
  "personRole": {
    "eq": "Owner"
  }
},
"component": {
  "name": "Radar",
  "version": "Automation"
}
}
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
X-API-Version: 1.0
Content-Type: application/json; charset=utf-8
[
  {
    "description": "Versions: 3, 4, 5, 6, 7, etc.",
    "id": 20496,
    "name": "Radar",
    "type": "Work Group",
    "isActive": true,
    "allowExternals": true
  },
  {
    "description": "Versions: 3, 4, 5, 6, 7, etc.",
    "id": 20499,
    "name": "RadarTest",
    "type": "Work Group",
    "isActive": true,
    "allowExternals": true
  }
]
```



## 14. SIGNING ON AND OFF

### 14.1 Sign-on to Radar with Single-Sign-On (SSO) Credentials

#### [A] Description

This API will be used to create a new Radar authentication session. Using any SSO credentials offered by IdMS and supported by Radar API, clients use this endpoint to initiate and retrieve the details of an Authentication session.

#### [B] Schedule

Required for version 1.5.1

#### [C] URL Scheme

GET /signon

#### [D] Response parameters

Key	Description	Data Type
accessToken	A user-specific and app-specific access token. The access_token is a string that represents the user's credentials for all Radar (REST) API calls.	String
tokenType	An attribute on the access token confirming the intended use of the token. All user authentication with Radar results in an Access Token of type "User".	String
tokenExpiresIn	Duration of time in seconds during which the Access Token is valid.	String

#### [E] Example

Client request (example uses SPNEGO for SSO Credentials):

```
GET /signon
WWW-Authenticate: Negotiate
```

Server response:

```
HTTP/1.1 200 OK
Status: 200
Content-Type: application/json;charset=UTF-8
{
    "accessToken" : "6959667c7889cac4aeedbd085efa27c84af8",
    "tokenType" : "User",
```



```
    "tokenExpiresIn" : "7200 Seconds"  
  }
```

## 14.2 Sign Off from Radar API

### [A] Description

This API destroys the authentication session with Radar. Any subsequent calls to a Radar API will not be able to successfully use the *access\_token*.

### [B] Schedule

Required for version 1.5.1

### [C] URL Scheme

```
POST /signoff
```

### [C] Request Attributes

No Request Body is expected.

### [D] Response Attributes

Key	Description	Data Type
accessToken	For confirmation on the client side, a copy of the access_token read from the HTTPS header and associated with the recently destroyed session.	Integer

### [E] Example

Client request:

```
POST /signoff  
Radar-Authentication: kjfkjaije;33902uj;js0033jksjij3
```

Server response:

```
HTTP/1.1 200 OK  
Status: 200  
{  
  "accessToken": "kjfkjaije;33902uj;js0033jksjij3"  
}
```