[Join Skype Meeting](https://meet.ericsson.com/yu.deng/QVRPDDW6)

https://jira-fe-p01.mr.ericsson.se:8443/browse/MF-63840?filter=11972

\\mrfiles2\userfiles

TLS RUN: move livetv

--test artifacts\build\tests\Search\SearchFromAllTVShows.js --env tlsprodc-chrome-win8 --var site=A --var slot=S76

**Remote Destop:**

**Subject:** Re: please allocate 2 VMS to China team

Please use:

TV3-AGNT-1614

TV3-AGNT-1615

user

mr\tvagent

12$TestAgent!

node node\_modules/nightwatch/bin/runner.js --test artifacts/build/tests/LiveTV/LiveTVGuideFilter10ft.js --env funk-chrome-10ft-1080

**Funk Test**: ---------------------------------------------------------->>>

Adult Store Test Case

config.json

"environment": "devtest",

"environment": "funk",

landing.json

"sts": "https://appgw-client-a.dev.mr.tv3cloud.com/S1/sts/",

"sts": "https://ottapp-appgw-client-a.funk.mr.tv3cloud.com /sts",

--------------------------------------------------------------------->>>

必须使用命令行：跑hydra

C:\EHOEZHU\ReachClient>setenv

C:\EHOEZHU\ReachClient>git request-whatif mf637523

git push origin users/ehoezhu/mf63752

git commit --m "#MF-63752" -m "package.json"

git merge origin/master

---------------------------------------------------------------------->>>

**Git submit：checkin**

-3. get your PR signed-off

-2. git fetch

-1. git merge origin/master

0. gulp,  gulp test

1. git checkout -b {new branch name} origin/master

{new branch name}:   users/emuyuan/bugs/123456

2. git merge --squash {old branch name}

{old branch name}: the branch that you used to publish the PR.

3. git commit -m "#123456" -m "description"

 git request-whatif [request-name].

4. git request-submit {request name} -b master

{request name}: NO white space.

5. look at ReachClient.Master.Submit

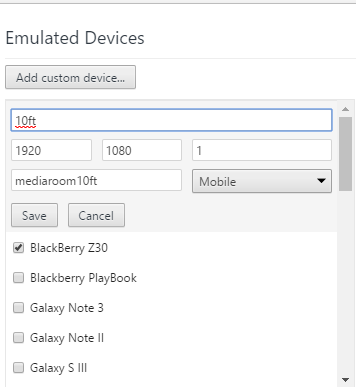
---------------------------------------------------------------------->>>

**OnCall:**

**https://conf-fe-p01.mr.ericsson.se:8443/pages/viewpage.action?spaceKey=EN&title=ReachClient+Oncall+Instruction**

---------------------------------------------------------------------->>>

<https://reachclient.funk.mr.tv3cloud.com/?forceStubPlayer=true> **10ft flashPlayer was different**

****

https://operatorportal.funk.mr.tv3cloud.com/workflow2/managefeaturegroups

---------------------------------------------------------------------->>>

Account:

|  |  |  |
| --- | --- | --- |
| [TV3PF\_LIYINGDAI@OUTLOOK.COM](mailto:TV3PF_LIYINGDAI@OUTLOOK.COM) |  | Tv3PF.Test |

|  |  |  |
| --- | --- | --- |
| [TV3PF\_JIASHUAN@OUTLOOK.COM](mailto:TV3PF_JIASHUAN@OUTLOOK.COM) |  | Tv3PF.Test |

---------------------------------------------------------------------->>>

**跑hydar**

git checkout -b {new branch name} origin/master

{new branch name}:   mf12345

2. git merge --squash {old branch name} users/exxx/mf123

{old branch name}: the branch that you used to publish the PR.

3. git commit -m "#MF-123456" -m "description"

4. git request-whatif {request name} mf12345

---------------------------------------------------------------------->>>

**权限申请：To:** Mohit Vadera <[mohit.vadera@ericsson.com](mailto:mohit.vadera@ericsson.com)>  
**Cc:** Yu Deng <[yu.deng@ericsson.com](mailto:yu.deng@ericsson.com)>  
**Subject:** fyi. request tfs access.

---------------------------------------------------------------------->>>

https://saucelabs.com/beta/login?redirect\_to=/beta/dashboard/tests

yudeng

159753

---------------------------------------------------------------------->>>

\\ 10.185.78.145

---------------------------------------------------------------------->>>

Git rename:

Branch Naming conventions：users/EID/mf{tasknumber}

Address: http://tfsmr:8080/tfs/IEB/MRGIT/\_git/ReachClient

---------------------------------------------------------------------->>>

Ericsson任务分发

<https://jira-fe-p01.mr.ericsson.se:8443/browse/MF-63752>

---------------------------------------------------------------------->>>

OnCALL

https://conf-fe-p01.mr.ericsson.se:8443/display/EN/ReachClient+Oncall+Instruction

npm test -- --test artifacts/build/tests/Dialogs/DetailDialogFilmstripChevron --env ci-chrome-win8

---------------------------------------------------------------------->>>

https://reachclient.funk.mr.tv3cloud.com/#settings

pdlomtv3pf@pdl.internal.ericsson.com

npm test -- --tag ci --tag funk

运行标有“CI”和“Funk”的所有测试

npm test -- --env devtest-ie10-win8 --tag P0

运行标有“P0”的所有win8和win10的系统环境

npm test -- --test artifacts/build/tests/LandingPage/LandingPageLogIn

运行一个环境

1.夜表只知道`artifacts / build`中的文件或文件，因此相应地在`tests`路径前缀。

2.测试文件扩展名应为`.js` - 或者您可以完全省略它，原因与上述相同。

`shield`捆绑Selenium，Chrome Webdriver和`--env`配置，在运行`nightwatch`之前启动Selenium服务器。

npm test -- --env devtest-localchrome

npm test -- --env ci-localchrome

Note: for running on the local browser the Chrome web browser and *\*Java JDK\** must be installed locally.

在本地浏览器上运行，javajdk必须安装

Hydra环境与其他环境不同，当您尝试对hydra进行测试时，不要忘记给出参数`--BuildNumber`。

npm test -- --env hydra-chrome-win8 --BuildNumber 2016rs-1783-12

Shield命令用于官方自动运行。 所有测试将被并行触发，所以本地平台在Shield中不受支持。

支持的参数：

**--env / --device --deployment**

指定测试运行的设备和环境。

npm run shield -- --deployment funk --device chrome-win8

npm run shield -- --env funk-chrome-win8

All available environment configurations: (N: Not gated / S: Ship(gated))

| | | hydra | ci | funk | sim | preprod |

| ------------------------ | ------------------------ |:-----:|:--:|:----:|:---:|:-------:|

| AndroidTabS | galaxy-s-tab | | N | N | s | s |

| chromeMountainLion | chrome-mountainlion | | S | S | | s |

| chromewin8 | chrome-win8 | S | S | S | s | s |

| FirefoxMac | firefox-Mac | | N | N | | s |

| FirefoxWin | firefox-win8 | | S | S | | s |

| IE11win8 | ie11-win8-saucelab | | S | S | s | s |

| ieEdge | ie-Edge | S | S | S | s | s |

| iPadAir | ipad-air | | N | S | s | s |

| iPadmini | ipad-mini | | S | S | s | s |

| IPTVSTB | iptv-ucstb | | S | S | n | n |

| Safari | safari | N | N | N | | s |

| UCSTB | 10ft-ucstb-monkeycagelab | | S | | | |

| 10ftEmulation | chrome-10ft-1080 | S | | N | | |

| 1ftEmulation | 1ft-ios-sim | S | | | | ` ` |

指定要使用的测试标签。 必填项目。 这不同于夜间的标签。 “，”支持多个标签，标签不区分大小写：

npm run shield -- --tag CI,Funk

排除测试用例运行。 可以使用全名指定多个测试用例，用'，'分隔：

npm run shield -- --tag CI --excludedCases FeedbackDialog,SearchRentPlayTVShow

输入内部编号，所以记者和日志都有使用测试的编号。

所有正式运行都需要BuildNumber。

npm run shield -- --deployment hydra --device chrome-win8 --BuildNumber "20150605hy-a20f04b-7"

运行所有测试并启用默认的自定义记录

npm run shield -- --reporter

只运行一个特定的测试文件并启用自定义记录

npm test -- -t [path to test file] --reporter [path to custom reporter file that can be found from defaults.json5]

e.g., npm test -- -t artifacts/build/tests/Settings/VerifySettingsPageElements --reporter artifacts/build/lib/custom-reporters/reporter.js

运行类型是测试报告中使用的值。 应从外部传递到盾构工程。

     可能的值：CIGating，CIProbation，MiniCI，TIP，PrivateRun，DailyTestRun，TestStabilityRun

     默认运行类型为PrivateRun。

`retry` defines amount of attempts to execute test again in case if it is failed.

npm run shield -- --deployment hydra --device chrome-win8 --BuildNumber "20150605hy-a20f04b-7" --retry 2

`testTimeout` defines timeout in milliseconds, after that time test case will be interrupted. Default time is 15 min.

npm run shield -- --deployment hydra --device chrome-win8 --BuildNumber "20150605hy-a20f04b-7" --retry 2 --testTimeout 600000

通过传递`--val key = value`这样的参数来引用或覆盖`configs / globals.js`中的全局配置。 可以重复参数以应用多个全局配置变量。 例子：

# Enable mock recording mode

npm run shield -- --var mockMode=record`

# Use another test account

npm run shield -- --var testAccountID=uxdesigntest@outlook.com --var testAccountPassword=95630MRUX$

注意：由于参数受命令行环境的限制，请注意如上所示使用单引号，如果值包含特殊的shell字符，如`$！＃<> |＆等等

注意：特定的环境也可以覆盖全局配置，也可以优先于`--var`变量。

多设备使用

----------------

对于运行需要多个设备的测试，已经创建了一个自定义运行程序，可以增加创建webdriver会话的功能

附加设备，并在同一测试中使用这些设备/会话。

另请参见lib / MultiDevice / readme.md

使用`--browserName`在特定设备上运行测试

The value of 'browserName' should be device GUID:

npm run multidevice -- --env ci-iptv-ucstb --test artifacts/build/tests/Authentication/LogInLogOut.js --browserName bd82d7e8-fdfa-4bc2-b492-31c49a231c84

npm test -- --env ci-iptv-ucstb --test artifacts/build/tests/Authentication/LogInLogOut.js --browserName bd82d7e8-fdfa-4bc2-b492-31c49a231c84

运行单个测试文件

npm run multidevice -- --env devtest-10ft-ucstb --test artifacts/build/tests/UC/StreamManagement/Multicast/MultideviceContentionResolution

**Run all tests in a folder**

Example:

npm run multidevice -- --env devtest-10ft-ucstb --group artifacts/build/tests/UC/StreamManagement/Multicast/

**Run multiple tests in shield**

Example:

npm run shield -- --multidevice --deployment devtest --device 10ft-ucstb --runType TestStabilityRun --tag UC --reporter --localrun



659/5000

＃＃＃ 一般  
小心选择你的测试用例。 实施核心测试场景。  
     一个。 Shield中的所有测试将用于每个环境（门控登录，CI，Funk）和TIP（在所有生产环境中）的官方自动化。  
    湾 如果测试不重要，请勿将其保存在Shield中。 即切割所有P2病例。  
  
\* CodeScript \*中的代码。 Shield编译所有`.ts`文件，并将所有`.js`，`.json`和`.json5`文件复制到`artifacts / build`，保留目录结构。  
     一个。 例外是lib / custom-commands。  
    湾 不要在.ts文件中直接使用“mstv”。 它导致编译错误。 在lib / custom-commands中编写单独的命令.js文件

测试用例

请查看现有的测试用例（推荐的Goldenthread测试），了解如何编写新的测试用例。

UI元素表示应该写在Page-Objects中。它们不应该在测试文件中硬编码。

2.每个测试文件都是具有多个步骤的一个测试场景，或包含一个区域的多个功能测试的测试套件。

    \*将同一测试文件中的此区域/场景所需的所有测试写入。例如，在browseMovie.ts中为浏览电影页面分组所有测试

    \*在一个测试文件中尽可能减少设置开销。官方自动化总是运行整个测试文件，其中步骤/测试依次执行。

    不要在以后的步骤/测试中重新设置，从以前的步骤/测试中恢复。例如，以前的测试去了browsemovie页面并测试了一些东西。不要回到你的网页。代替

    继续测试其他情况下的浏览页面。

    \*大写第一个单词的测试步骤名称，不要使用大写为其他单词，除了首字母缩略词。

3.尝试为1ft，2ft和所有浏览器进行测试工作。减少代码重复。

    一个。定义和使用HTML / CSS选择器有效地套装1ft和2ft用户代理。

    湾为10ft创建单独的测试，自定义命令和页面对象，因为UI是不同的，10ft应该使用按键

4.10ft测试应使用按键操作，而不是使用鼠标点按动作

5.在测试脚本中有效地使用页面对象。定义可以基于识别的用户代理执行鼠标点击或按键的方法。

示例：在\ Shield \ lib \ page-objects \ YouHub.js下goToYouMightLikeFeed（）可以用作参考

7.新的测试用例必须遵循用于生成测试用例文档的注释格式。

    \*开始时提出意见，包括测试用例名称，每个测试步骤的说明和验证。

    \*全局变量必须隐藏在测试用例文档中，在变量的顶部留下`/ \*\* @hidden \* /`。

    \*评论中测试步骤的名称必须与代码中的测试步骤名称相匹配。

    \*更新'docs / sdk / tsconfig4tests.json'以包含新的测试用例的路径。

\*更新“Shield.njsproj”以包含新的测试用例的路径。

这些是我们在自动化中使用的当前测试标签。

     - 官方自动化中使用的官方标签 -

     - 不要用这些标签标记你的测试，除非他们准备好进行官方的自动化 -

    环境相关标签

    CI：测试将遍布各地。 CI，Funk，Preprod，Sim，门控登录（所有可用的浏览和所有环境的设备）和TIP

    Funk：测试将在任何地方运行CI，Funk，Preprod，Sim，门控登录（所有可用的浏览和所有环境的设备）和仅在Chrome上的TIP

    CIProbation：仅在门控检查中运行的测试

    浏览器和设备相关标签

    10英尺

    1ftEmu

    Android2ft

    苹果浏览器

    iPad的

    firefoxMac

    模块相关标签

    GT：金线测试。

    PBR：PBR测试。

    封闭式：

    TELUS：仅在TELUS环境中运

     - 官方自动化使用的官方标签 -

    BVT1ft：仅在1ft运行的核心测试

    BVT10ft：仅在10ft运行的核心测试

     - 没有支持 -

    P0，P1，P2：所有测试完成后，我们不支持任何P0，P1和P2标签。 Shield的所有测试都打算在官方自动化中运行。

页面对象

lib / common / selectors.js不再受支持，因为它太大了。 改用特定页面对象

2.页面对象应该与客户端代码中的UI结构相匹配。 每个页面或每个组件应该有自己的页面对象以避免重复。

     例如，

     一个。 设置按钮，媒体室图标在MenuBarComponent中

    湾 FeedBackDialog有自己的页面对象

10ft的情况需要自己的页面对象文件