Details

Below spreadsheet contains the serial details for Tellus along with the purpose of allocation

Individual_
201769

Please update the spreadsheet if you reassign the serial numbers and add new serials

Convert Serial no's to command line

Thursday, June 08, 2017   12:00 AM

Here is the rpd filed for the same: http://is.factset.com/rpd/summary.aspx?messageid=27882084

Steps:
1) Ensure you have access to the location: //depot/appqa/bpo/online/" in appqa-dev group
2) If you don't have add yourself to this group using: http://p4ws.factset.com/perforce-ws/
3) The file: **build_chooser_sn_file.txt** would have the serial number entries with the following details:
   a. **Serial Number** - this is the serial number of the user you'd like to use build_chooser with
   b. **Build Choice** - this will represent a stage that you'd like to use. You'll notice that most of these are set to "LIVE" but "DEVEL" and "QA" are also valid choices here.
   c. **HTML Choice** - this is stage of the *HTML* build that would be used: nearly all currently list "PRODUCTION" however "STAGING" and "DEVEL" are also valid options.
   d. **CACCESS** - this will set the CACCESS file to be used. I'm not as well versed in all the different options, but you can see plenty of examples of how other accounts are set up.
   e. **HTML OS** - this is also a little bit outside of my expertise but from what I can see this has something to do with which platform runs/accesses the HTML build: in all cases this is set to "DEFAULT" and I doubt you'll need to supply any other option.
4) If all the above values are provided followed by a serial number, when you launch a serial number it would use the values provided in this file
   To force a serial number to use the cmd line mode, simply apply a "!" followed by a space and a comment. Here's an example:
   366947 ! switch to cmd line, RPD:17227069
5) Submit the file to P4 once you make changes to build_chooser_sn_txt file.
6) Submitting to Perforce does not automatically update the file on disk, please follow the below steps to update the file on disk
   a. Login to your inhouse account or FDSQAR_C account from FactSet terminal window

b. Once you login successfully, It would show the node you are connected to as shown below:

```
FDSC11::"UNAKKA"   Wednesday, June 07, 2017, 14:26:24 (2:26:24 PM)

Setting display to node 164.55.41.238
NJ C11 S7:[UNAKKA]$ syncpass
Synching password to all VMS clusters. You'll be notified on this terminal when complete.
NJ C11 S7:[UNAKKA]$
Reply received on FDSC11 from user SYNCPASS at FDSC04 Batch   14:27:12
Password ( 7-JUN-2017 14:26) sync from FDSC successful to FDSA,FDSB,FDSD

NJ C11 S7:[UNAKKA]$ @PRD_PROGS:[QA_TOOLS]PUSH_BUILD_CHOOSER.COM
Job BUILD_CHOOSER_SYNC (Entry 377) scheduled
  OTHER_CLUSTER = "FDSA"
Job BUILD_CHOOSER_SYNC (Entry 8930) scheduled
  OTHER_CLUSTER = "FDSB"
Job BUILD_CHOOSER_SYNC (Entry 12452) scheduled
  OTHER_CLUSTER = "FDSD"
Job BUILD_CHOOSER_SYNC (Entry 1368) scheduled
NJ C11 S7:[UNAKKA]$ _
```

c. Ensure you are connected to FDSC node to execute the below command.

d. **@PRD_PROGS:[QA_TOOLS]PUSH_BUILD_CHOOSER.COM**
Note: this command will sync EVERYTHING in "//depot/appqa/bpo/online/" to disk on all 4 clusters (FDSA, FDSB, FDSC, FDSD). This generally isn't an issue, since most of the files in that location do not change but it does mean that you should not submit changes here (the Perforce depot) unless you intend for them to be put on disk immediately.

7) Deploy a VM and check if the serial number is converted to accept build chooser options as a String or not.

8) Hope it helps!

Other notes:
1) When adding a new serial number to the list, we put it in ascending numerical order to make things easier to find and keep things organized.

2) Supplying the RPD in the comment (the portion after the "!" helps tie the change back to the originating request, however, it's also important to mention the RPD in the Perforce submit comment. Here's an example:

   Per info in RPD:11926405 100 S/Ns set to Cmd In-
   427224->427253 (30)
   427412->427441 (30)
   427743->427772 (30)
   427814->427823 (10)

3) Use it only for converting a Tellus serial number to string command. If there is a request from anyone please have an rpd filed for the same before you convert.

If you don't have the inhouse password you need to file an rpd adding the product "Password - VMS". Here is the sample rpd filed: RPD:31810667.
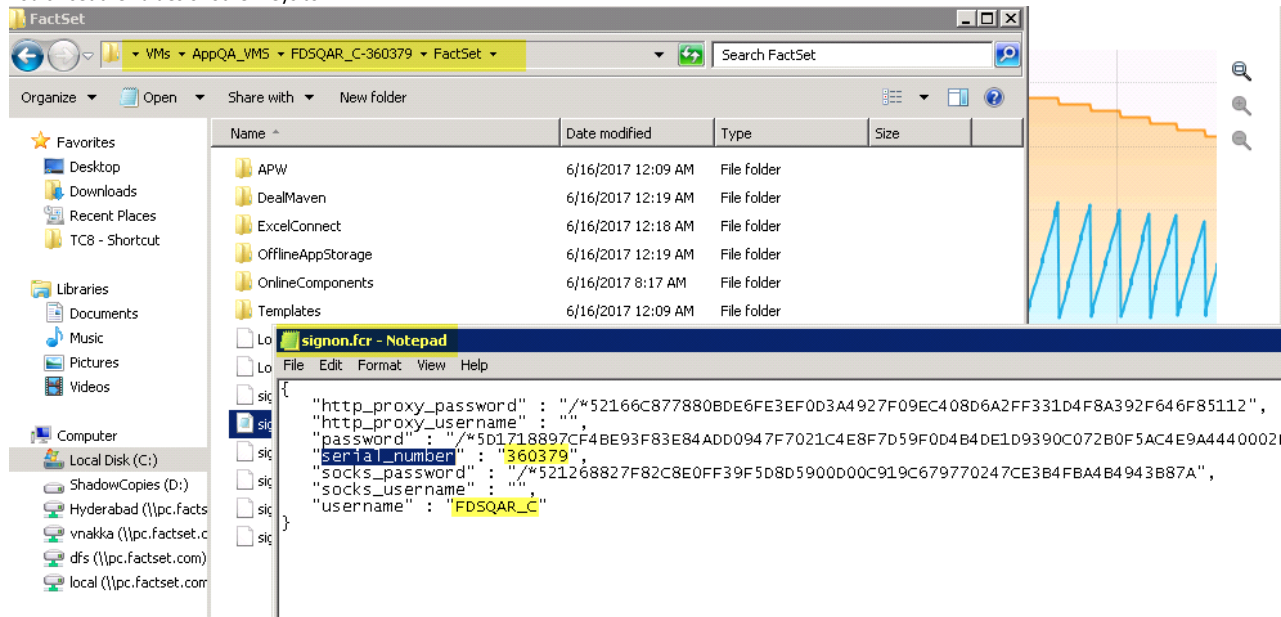
Note: username is your windows user account

Auto-validate for first time

Friday, June 16, 2017   8:28 PM

Create the serial number directories
Once you have the serial numbers converted and ready to use please follow these steps:

- Identify the serial numbers that needs to be added to Tellus server and logon to the appropriate Server
- Create the serial number directories on the servers at C:\VMs\AppQA_VMS by following the below steps:
    - Create a directory with the naming convention: FDSQAR_C-**#SerialNo**
        - #SerialNo is the new serial number you are going to add
    - Once you create a directory cd to it and create a directory named: **FactSet**
    - Inside the FactSet directory we need to create a file named signon.fcr
    - For this you can copy it from any of the serial number directories for ex: FDSQAR_C-360379/FactSet/
    - Open signon.fcr and edit the value for "serial_number" to the new **#SerialNo** and **"username"** to **"FDSQAR_C"**
    - You cn set the values of other keys to ""

If you need to create more serial number directories, I have created a python file that would create the serial number directories along with the signon.fcr files. You can find it at C:\VMs\scripts with the name: CreateSerNoDir.py.

Please follow the below steps to use this file:
- Change the username to the appropriate one if it is not 'FDSQAR_C'
- Change the values in range to the range of serial numbers in the below line:

```
28  def main():
29      username = 'FDSQAR_C'
30      filename = 'signon.fcr'
31
32      #change the range of serials below
33      for i in range(683222  ,683223):
34          # change the directory to loc where serial num direcories are present
35          os.chdir('C:\\VMs\\AppQA_VMS')
36          #check whether we are at correct location
37          if os.getcwd()=='C:\\VMs\\AppQA_VMS':
38              #create the serial number i.e. FDSQAR_C-serNo directory
39              #print i
40              os.mkdir('%s-%s' %(username, i))
41              # change the current directory to the serial num dir that we created
42              curDir = os.getcwd()+'\\'+'%s-%s' %(username, i)
43              os.chdir(curDir)
44              #create directory named Factset under FDSQAR_C-SerNo
```

- Executing the above file would create the directories with name: FDSQAR_C-1 till FDSQAR_C-4. You can change the range if your serial number list is in sequence.
- If it is not in sequence for ex: 425671, 56891, 2456145, 6245788 use this steps:
    - Create a list of serial numbers and store them in the list format as: serials = [425671, 56891, 2456145, 6245788]
    - Once the list is created replace the **range()** with the name of the list.
- Execute the python file and it creates the directories for you
- Note: Before trying it on a Tellus server try to execute it locally on your computer to see how it works.

- Yayyy! the serial number directories are created :)

Autovalidate the serial numbers on Workstation
- To Do - Avanthi Rentala

Serial num : 686366 'automation.tc6863651@factset.net'

Tellus Serial Number API Documentation

Monday, October 02, 2017   11:40 AM

TellusCI-API ver 1.0 has been deployed to **TellusStgB01** machine which will act as the production environment for the TellusCI-API. As of now this API will be able to perform only the following.
1. Block specific or random serialnumbers
    a. Check availability for the requested serial number and other resources mentioned in the request
    b. Clear the FDSAV2 Stage and FDSAV2 Overrides
    c. Block the serial number along with it set the fdsav2stage, fdsav2overides and caccess if mentioned
2. Releasing the serial number
    a. Validate the request
    b. Clear the FDSAV2 Stage and FDSAV2 Overrides
    c. Release the serial number

URLs:
Request: http://tellusstgb01:8080/tellusci/ci/request
Release: http://tellusstgb01:8080/tellusci/ci/release

Following is the format for the request:

```
{
    "RequestJson": {
        serialNumber:
        [vmsusername]:
        [poolname]:
        [caccess]:
        [fdsav2stage]:
        [fdsav2overides]:
    }
}
```

where serialnumber is mandatory(value should be either should be serialnumber/"random") keys mentioned in the [] are optional and will be defaulted to the following:
vmsusername: FDSQAR_C
poolname: CID
caccess: LARGESETCACCESS
fdsav2stage/fdsav2overides: blank

Example request for CID:
```
{
  "RequestJson": {
    "serialnumber": "random",
    "vmsusername": "FDSQAR_C",
    "fdsav2stage": "edge",
    "fdsav2overrides": "review_universal-screening_version-0-1_2248"
  }
}
```

Example:
```
{
  "RequestJson": {
    "serialnumber": "random",
    "vmsusername": "FDSQAR_C",
    "fdsav2stage": "edge",
    "fdsav2overrides": "review_universal-screening_version-0-1_2248"
  }
}
```

Response for the request will be

a. For positive response:
   Status code: 200
   Response type: JSON

```
   {
     "ResponseJson": {
         serialnumber:
         requestId:
         success:
       }
   }
```
   where
         1. serialnumber : serialnumber which was blocked
         2. requestId: Requestid generated for that particular request
         3. success: Success message for the request made

b. For negative response:
   Status code: 400/500
   Response type: JSON

```
   {
     "ResponseJson": {
          error:
         requestId:
       }
   }
```
    where
          1. requestId: Requestid generated for that particular request
          2. error: Error message for the request made

Release:

```
 {
"RequestJson": {
        serialnumber: <NUM>
       }
}
```
     Where serial number is mandatory and the value should be a serial number which needs to be released(in the release request, serialnumber cannot be 'random'). Rest of the fields from the Request can be given but they will be ignored.

Response for the release will be
a. For positive response:
   Status code: 200
   Response type: JSON

```
   {

     "ResponseJson": {

         serialnumber:

         requestId:

         success:

       }

   }
```
   where
         1. serialnumber : serialnumber which was released
         2. requestId: Requestid generated for that particular request
         3. success: Success message for the release made

b. For negative response:

Status code: 400/500
Response type: JSON

```
{
  "ResponseJson": {
      error:
      requestId:
    }
}
  where
        1. requestId: Requestid generated for that particular request
        2. error: Error message for the request made
```

Detailed documentation on usage and examples has been mentioned on the following OneNote:

As of now we have decided the following to be part of next release for CI-API
1. Support for requesting multiple serialnumbers with comma separated values
2. Support for release request to be honored for a serial number only if requested by the blocker
3. Looking into and Fixing Machine name lookup logic for logging requester machine name
4. Ability to set Caccess, FDSAV2Session for any requested serial number

Examples

HTTP POST
Content-Type: application/json
Accept: application/json

Example Request:
```
{"RequestJson":{"serialnumber":686410}}
```

Note: serialnumber can be number or string.

For any serial number:
```
{"RequestJson":{"serialnumber":"random"}}
```

Requesting from a specific pool:
```
{
  "RequestJson":{
    "serialnumber":"random/<serialnumber>",
    "poolname ":"<name of the pool>"
  }
}
```

The given pool name should be valid and this will be validated from apiconfig.properties file. For now in TellusDev only takes FDSQAR_C-QATesting and TellusStgB01 only takes CID for valid poolnames

Pools for Protractor and Ruby: We currently don't have separate pools for Protractor and Ruby.

Example response:
For 'request':
```
{"ResponseJson": {
    "requestId": 2017100211342298305,
    "serialnumber": 686410,
    "success": "Successfully blocked serial number"
}}
```

For release: "success": "Successfully released serial number"

Example error response:

Return HTTP Code 500 Server Error

Response body for `request`:
```
{
    "ResponseJson": {
        "requestId": 2017100211393075176,
        "error": "Requested serial number does not exist"
}
}
```

Example error for `release`:
"error": "Requested serial number is not avaialbe for release"

Serial Number API request examples

Monday, January 22, 2018   11:10 AM

Request Method: POST
Request URL: http://tellusstgb01:8080/tellusci/ci/request
Request Headers:
      Content-Type: application/json
      Accept: application/json

Example - Request any serial number:

```
{"RequestJson":{"serialnumber":"random"}}
```

Example - Request any serial number with override:

```
{
  "RequestJson": {
    "serialnumber": "random",
    "vmsusername": "FDSQAR_C",
    "fdsav2stage": "edge",
    "fdsav2overrides": "review_universal-screening_moc-breakautomation_2426"
  }
}
```

Example - Request any serial number in specific pool:

```
{"RequestJson":{"serialnumber":"random", "poolname":"CID"}}
```

Example - Release serial number:

```
{"RequestJson": {"serialnumber": 686363}}
```