## FOTA Capabilities (Wi-Fi)

1. FOTA over HTTP/HTTPS
2. Image download from Cloud or any HTTP/web server
3. Two copy solution. Backup copy of the correct firmware always exists.
4. Image integrity check using sha256 hash.
5. Error handling and recovery:

If any error occurs during downloading the image or updating the configuration files (part.json/boot.json/fota\_config.json), the device will remain in the current image. If a reboot happens (due to issues like power failure) during an image download or configuration files upgrade, the device will boot with the current image.

1. JSON based configuration

Have a fota\_config.json file in the device. This file will be present in the AT command application package (binaries/product/at/root\_fs/root). This file will have the necessary information to fetch an updated (new) version of the fota\_config.json file in the server. Hence, to start with, there will be two fota\_config.json files. One that is in the device rootfs is referred to as the local copy, the other at the server is referred to as the remote copy.

The remote fota\_config.json in the sever will have bumped-up versions of package\_version and version under firmware. These versions in the remote config file will be more than that in the local config file.

The remote fota\_config.json file at the server will have necessary information to download the firmware.

|  |
| --- |
| {  "package\_version" : "3.0",  "files" : [  {  "type" : "configuration",  "name" : "fota.config",  "protocol" : "http",  "hostname" : "192.168.1.202",  "port" : 80,  "secured" : 0,  "uri" : "xxx/fota\_config\_new.json"  },  {  "type" : "firmware",  "name" : "atcmd",  "version" : "3.0",  "protocol" : "http",  "hostname" : "192.168.1.202",  "port" : 80,  "secured" : 0,  "uri" : "/xxx/t2\_atcmds.elf"  }  ]  } |

On booting, after at+fota=1 is issued, the device refers to the configuration section of the local config file, downloads the new configuration file from the server and checks if an updated package is available. Firmware downloaded in a higher version package is available.

## Triggering FOTA Operation

After the Wi-Fi connection is established, (at+wcon=ssid,passphrase), execute at+fota=1 command. This will trigger the FOTA.

## Image Integrity Using Hash

If hash is present for firmware, in the remote fota\_config.json, then image integrity is checked.

1. Creating hash (in ubuntu VM): $sha256sum <elf file>

For example:

|  |
| --- |
| Freertos\_sdk\_x.y/binaries/product/at/bin$ sha256sum t2\_atcmds.elf  0200ac047cf0bff69d9b71a12144d1bec088aad865bb17484605f305709fd1f2 t2\_atcmds.elf |

1. Copy the output string to fota\_config\_new.json to update the hash. Following is a remote fota\_config.json with hash:

|  |
| --- |
| {  "package\_version" : "3.0",    "files" : [  {  "type" : "configuration",  "name" : "fota.config",  "protocol" : "http",  "hostname" : "192.168.1.202",  "port" : 80,  "secured" : 0,  "uri" : "/**xxx**/fota\_config\_new.json"  },  {  "type" : "firmware",  "name" : "atcmd",  "version" : "3.0",  "protocol" : "http",  "hostname" : "192.168.1.202",  "port" : 80,  "secured" : 0,  "uri" : "/xxx/t2\_atcmds.elf"  "hash" : "0200ac047cf0bff69d9b71a12144d1bec088aad865bb17484605f305709fd1f2"  }  ]  } |

## JSON File Validation

**Note**:

1. No blank lines should be present in the json files.
2. Ensure that any json file is validated after any change to it using offline/online tools. One of the online tools to accomplish this is: <http://json.parser.online.fr/>.