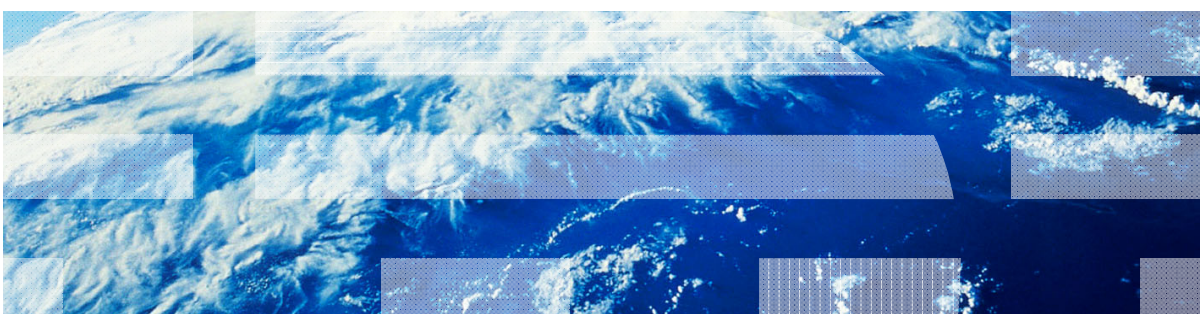
IBM Worklight V5.0.5
Getting Started

Module 7.8 – Encrypted Cache



18 January 2013

Trademarks

- IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Worklight is a trademark or registered trademark of Worklight, an IBM Company. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.
- Other company products or service names may be trademarks or service marks of others.
- This document may not be reproduced in whole or in part without the prior written permission of IBM.

About IBM®

- See http://www.ibm.com/ibm/us/en/

Agenda

- What is encrypted cache?
- Supported browsers and devices
- Create and Open
- Read, Write, and Remove
- Close and Destroy
- Change encryption key
- Exercise

What is encrypted cache?

- Encrypted cache is a mechanism for storing sensitive data on the client side
- Encrypted cache is implemented by using HTML5 web storage technology, which allows data to be saved locally and retrieved on subsequent application use or relaunch
- Data is encrypted with a combination of user-provided key and server-retrieved randomly-generated token, which makes it more secure
- Data is stored using key-value pairs
- Encrypted cache is like a security deposit box – it remains open until you close it, so remember to close the cache when you have finished working with it

Agenda

- What is encrypted cache?
- Supported browsers and devices
- Create and Open
- Read, Write, and Remove
- Close and Destroy
- Change encryption key
- Exercise

Supported browsers and devices

- Encrypted Cache is implemented using HTML5 web storage technology
- Mobile devices HTML5 web storage support chart

Show all versions	iOS Safari	Opera Mini	Opera Mobile	Android Browser	
3 versions back	3.2				
2 versions back	4.0-4.1		10.0	2.1	
Previous version	4.2-4.3		11.0	2.2	
Current	5.0	5.0-6.0	11.1	2.3	3.0
Near future				4.0	
Farther future					

= Supported = Not supported

- For additional information, see http://caniuse.com

Agenda

- What is encrypted cache?
- Supported browsers and devices
- Create and Open
- Read, Write, and Remove
- Close and Destroy
- Change encryption key
- Exercise

Creating and opening encrypted cache

- To create or open previously created encrypted cache use the following API:
  - **WL.EncryptedCache.open(credentials, createIfNone, onComplete, onError);**
    - credentials – string value representing user-provided password
    - createIfNone – Boolean value specifying whether new encrypted cache should be created if none is found
    - onComplete – a callback function to be invoked when cache opening/creating is complete
    - onError - a callback function to be invoked when cache is not successfully opened/created.

```
WL.EncryptedCache.open(key, true, onOpenComplete, onOpenError);
function onOpenComplete(status){
    alert("Encrypted cache succesfully opened");
}
```

  - Note that the application must be able to connect to Worklight® server in order to create a new encrypted cache

Creating and opening encrypted cache

- A callback function can receive one of the following statuses:
  - **WL.EncryptedCache.OK** – Encrypted cache was successfully opened or created
  - **WL.EncryptedCache.ERROR_CREDENTIALS_MISMATCH** – an attempt was made to open existing encrypted cache using wrong credentials
  - **WL.EncryptedCache.ERROR_SECURE_RANDOM_GENERATOR_UNAVAILABLE** – unable to generate random token due to Worklight® Server unavailability
  - **WL.EncryptedCache.ERROR_NO_EOC** – could not open encrypted cache because it was not previously created
  - **WL.EncryptedCache.ERROR_LOCAL_STORAGE_NOT_SUPPORTED** – device does not support HTML5 local storage
  - **WL.EncryptedCache.ERROR_KEY_CREATION_IN_PROGRESS** – an open() or changeCredentials() request is already running

Creating and opening encrypted cache

```
WL.EncryptedCache.open(key, true, onOpenComplete, onOpenError);
function onOpenComplete(status){
    alert("Encrypted cache succesfully opened");
}
function onOpenError(status){
    busyIndicator.hide();
    switch(status){
        case WL.EncryptedCache.ERROR_KEY_CREATION_IN_PROGRESS:
            alert("ERROR: KEY CREATION IN PROGRESS");
            break;
        case WL.EncryptedCache.ERROR_LOCAL_STORAGE_NOT_SUPPORTED:
            alert("ERROR: LOCAL STORAGE NOT SUPPORTED");
            break;
        case WL.EncryptedCache.ERROR_NO_EOC:
            alert("ERROR: NO EOC");
            break;
        case WL.EncryptedCache.ERROR_COULD_NOT_GENERATE_KEY:
            alert("ERROR: COULD NOT GENERATE KEY");
            break;
        case WL.EncryptedCache.ERROR_CREDENTIALS_MISMATCH:
            alert("ERROR: CREDENTIALS MISMATCH");
            break;
    }
}
```

Agenda

- What is encrypted cache?
- Supported browsers and devices
- Create and Open
- Read, Write, and Remove
- Close and Destroy
- Change encryption key
- Exercise

Reading and writing data with encrypted cache

- When the encrypted cache is open, you can perform operations on it such as reading, writing and removing data
- To store data in encrypted cache use the following API:
  - WL.EncryptedCache.write(credentials, value, onSuccess, onFailure);

```
WL.EncryptedCache.write(key, value, onWriteSuccess, onWriteFailure);
function onWriteSuccess(status){
    alert("Succesfully encrypted into cache.");
}
function onWriteFailure(status){
    if (status == WL.EncryptedCache.ERROR_EOC_CLOSED)
        alert("Encrypted cache closed, write failed. error code= "+ status);
}
```

Reading and writing data with encrypted cache

- To read data from the encrypted cache use the following API:
  - WL.EncryptedCache.read(credentials, onSuccess, onFailure);

```
WL.EncryptedCache.read(key, onDecryptReadSuccess, onDecryptReadFailure);
function onDecryptReadSuccess(value){
    alert("Read success. Retrieved value :: " + key + " = " + value);
}
function onDecryptReadFailure(status){
    alert("Encrypted cache closed, reading failed");
}
```

- To remove data from the encrypted cache use the following API:
  - WL.EncryptedCache.remove(key, onSuccess, onFailure);

```
WL.EncryptedCache.remove(key, onRemoveSuccess, onRemoveFailure);
function onRemoveSuccess(status){
    alert("Succesfully removed from cache.");
}
function onRemoveFailure(status){
    alert("Encrypted cache closed, remove failed");
}
```

Agenda

- What is encrypted cache?
- Supported browsers and devices
- Create and Open
- Read, Write, and Remove
- Close and Destroy
- Change encryption key
- Exercise

Closing and destroying encrypted cache

- To avoid possible undesired access to encrypted cache, close it
- After encrypted cache is closed, access to its data is not possible without the encryption key that was used to create it
- Use the following API to close the encrypted cache
  - WL.EncryptedCache.close(onComplete, onFailure);

```
function closeCacheClicked(){
    WL.EncryptedCache.close(onCloseCompleteHandler, onCloseFailureHandler);
}
function onCloseCompleteHandler(status){
    alert("Encrypted cache closed successfuly");
}
function onCloseFailureHandler(status){
    alert("Could not close Encrypted cache");
}
```

Closing and destroying encrypted cache

- Encrypted cache can be wiped from the local storage
- After encrypted cache is destroyed there is no way to return the data that was stored in it
- Destroy encrypted cache only if you are sure that data stored in it will never be required again, or as a last measure if the encryption key is lost
- To destroy an encrypted cache use the following API:
  - WL.EncryptedCache.destroy(onComplete, onError);

```
function destroyCacheClicked(){
    WL.EncryptedCache.destroy(onDestroyCompleteHandler, onDestroyErrorHandler);
}
function onDestroyCompleteHandler(status){
    alert("Encrypted cache destroyed");
}
function onDestroyErrorHandler(status){
    alert("Error destroying Encrypted cache");
}
```

Agenda

- What is encrypted cache?
- Supported browsers and devices
- Create and Open
- Read, Write, and Remove
- Close and Destroy
- Change encryption key
- Exercise

Change encryption key

- While encrypted cache is in the open state, it is possible to change the encryption key
- To do so, use the following API:
  - **WL.EncryptedCache.changeCredentials(credentials, onComplete, onError)**
    - **credentials** – new user password to be used.
    - **onComplete** – a callback function to be invoked when complete.
    - **onError** – a callback function to be invoked in case of an error.
- Callback receives a status object with same structure as WL.EncryptedCache.open()

Agenda

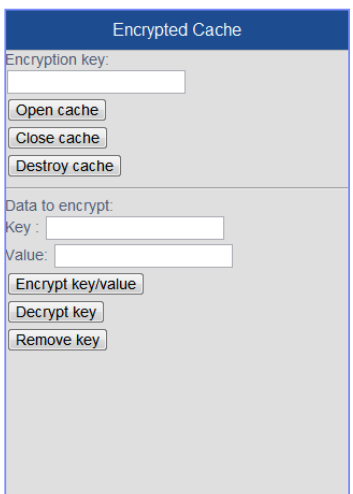
- What is encrypted cache
- Supported browsers and devices
- Create and Open
- Read, Write, and Remove
- Close and Destroy
- Change encryption key
- Exercise

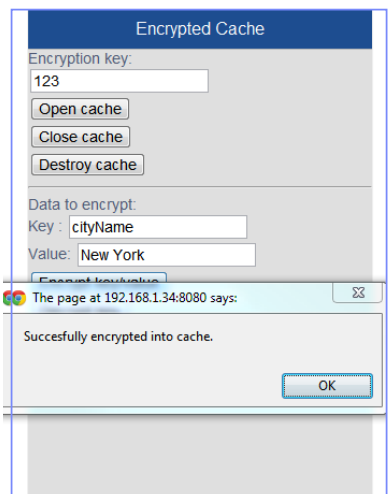
Exercise

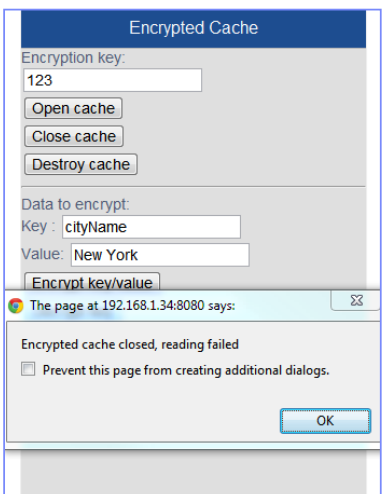
- Create an application that performs the following functions:
  - Creates an encrypted cache with a user-provided encryption key
  - Stores some key-value pair data in it
  - Closes the encrypted cache
  - Tries to access encrypted data while cache is in closed mode
  - Tries to open encrypted cache with an invalid encryption key
  - Opens encrypted cache with the correct encryption key
  - Retrieves previously stored data from encrypted cache
  - Closes encrypted cache
  - Destroys encrypted cache

Exercise

- The sample for this training module can be found in the Getting Started page of the IBM Worklight documentation website at http://www.ibm.com/mobile-docs







Check yourself questions

- Connectivity to Worklight server is required only in order to:
  - Create a new encrypted cache
  - Open an existing encrypted cache
  - Read and write values to encrypted cache
  - Destroy encrypted cache
- Which of the following APIs is synchronous and does not require callbacks to be set up?
  - WL.EncryptedCache.open
  - WL.EncryptedCache.read
  - WL.EncryptedCache.destroy
  - All encrypted cache APIs are asynchronous and require setting up callbacks for success and failure
- Which of the following sentences correctly describes the encrypted cache?
  - Encrypted cache is stored in the device native storage. Its size is limited by the free space on a device, therefore large amounts of data can be stored.
  - HTML5 WebStorage is used for storing encrypted cache; therefore the amount of data stored in it is limited to several megabytes
  - Encrypted cache is stored on Worklight server. Its size is limited by the free space in the Worklight Server database, therefore large amounts of data can be stored
  - Encrypted cache is stored in virtual memory. Its size is limited by the device RAM and it is erased each time the user quits the application.

Check yourself questions

- Connectivity to Worklight server is required only in order to:
  - Create a new encrypted cache
  - Open an existing encrypted cache
  - Read and write values to encrypted cache
  - Destroy encrypted cache
- Which of the following APIs is synchronous and does not require callbacks to be set up?
  - WL.EncryptedCache.open
  - WL.EncryptedCache.read
  - WL.EncryptedCache.destroy
  - All encrypted cache APIs are asynchronous and require setting up callbacks for success and failure
- Which of the following sentences correctly describes the encrypted cache?
  - Encrypted cache is stored in the device native storage. Its size is limited by the free space on a device, therefore large amounts of data can be stored.
  - HTML5 WebStorage is used for storing encrypted cache; therefore the amount of data stored in it is limited to several megabytes
  - Encrypted cache is stored on Worklight server. Its size is limited by the free space in the Worklight Server database, therefore large amounts of data can be stored
  - Encrypted cache is stored in virtual memory. Its size is limited by the device RAM and it is erased each time the user quits the application.

Notices

- Permission for the use of these publications is granted subject to these terms and conditions.
- This information was developed for products and services offered in the U.S.A.
- IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.
- IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
  - IBM Director of Licensing
    IBM Corporation
    North Castle Drive
    Armonk, NY 10504-1785
    U.S.A.
- For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:
  - Intellectual Property Licensing
    Legal and Intellectual Property Law
    IBM Japan Ltd.
    1623-14, Shimotsuruma, Yamato-shi
    Kanagawa 242-8502 Japan
- **The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law**: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.
- This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.
- Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.
- IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.
- Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:
  - IBM Corporation
    Dept F6, Bldg 1
    294 Route 100
    Somers NY 10589-3216
    USA
- Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.
- The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.
- Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

COPYRIGHT LICENSE:

- This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.
- Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

  © (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

Support and comments

- For the entire IBM Worklight documentation set, training material and online forums where you can post questions, see the IBM website at:
  - http://www.ibm.com/mobile-docs
- **Support**
  - Software Subscription and Support (also referred to as Software Maintenance) is included with licenses purchased through Passport Advantage and Passport Advantage Express. For additional information about the International Passport Advantage Agreement and the IBM International Passport Advantage Express Agreement, visit the Passport Advantage website at:
    - http://www.ibm.com/software/passportadvantage
  - If you have a Software Subscription and Support in effect, IBM provides you assistance for your routine, short duration installation and usage (how-to) questions, and code-related questions. For additional details, consult your IBM Software Support Handbook at:
    - http://www.ibm.com/support/handbook
- **Comments**
  - We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this document. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.
  - For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.
  - When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state.
  - Thank you for your support.
  - Submit your comments in the IBM Worklight forums at:
    - https://www.ibm.com/developerworks/mobile/mobileforum.html
  - If you would like a response from IBM, please provide the following information:
    - Name
    - Address
    - Company or Organization
    - Phone No.
    - Email address

Thank You