

Silk Test 15.0

Testing Mobile Web
Applications

Micro Focus
575 Anton Blvd., Suite 510
Costa Mesa, CA 92626

Copyright © Micro Focus 2014. All rights reserved. Portions Copyright © 1992-2009 Borland Software Corporation (a Micro Focus company).

MICRO FOCUS, the Micro Focus logo, and Micro Focus product names are trademarks or registered trademarks of Micro Focus IP Development Limited or its subsidiaries or affiliated companies in the United States, United Kingdom, and other countries.

BORLAND, the Borland logo, and Borland product names are trademarks or registered trademarks of Borland Software Corporation or its subsidiaries or affiliated companies in the United States, United Kingdom, and other countries.

All other marks are the property of their respective owners.

2013-12-09

Contents

Testing Mobile Web Applications	4
Clients for Mobile Web Testing	4
Testing Mobile Web Applications on Android	4
Testing Mobile Web Applications on a Physical Android device	4
Testing Mobile Web Applications on an Android Emulator	5
Installing a USB Driver	5
Enabling USB-Debugging	6
Manually Setting the Open Agent as a Proxy for an Android Device or Emulator	6
Configuring the Android Emulator for Silk Test	7
Recording Mobile Web Applications	8
Interacting with a Mobile Device	9
Troubleshooting when Testing Mobile Web Applications	9
Manually Adding a Root Certificate to Test a Secure Web Application	11
Limitations for Testing Mobile Web Applications	12

Testing Mobile Web Applications

Silk Test enables you to automatically test your mobile Web applications. Automated testing with Silk Test provides the following benefits:

- It can significantly reduce the testing time of your mobile Web applications.
- You can create your tests once and then test your mobile Web applications on a large number of different devices and platforms.
- You can ensure the reliability and performance that is required for enterprise mobile Web applications.
- It can increase the efficiency of QA team members and mobile Web application developers.
- Manual testing might not be efficient enough for an agile-focused development environment, given the large number of mobile devices and platforms on which a mobile Web application needs to function.

Clients for Mobile Web Testing

The following Silk Test clients support mobile Web testing:

- Silk Test Workbench
- Silk4J
- Silk4NET

Testing Mobile Web Applications on Android

Silk Test enables you to test a mobile Web application on an Android device or an Android emulator.

Testing Mobile Web Applications on a Physical Android device

To test a mobile Web application on a physical Android device, perform the following tasks:

1. Connect the device to the machine on which Silk Test is installed.
2. If you are testing this Android device for the first time on this machine, install the appropriate Android USB Driver on the machine.

For additional information, see *Installing a USB Driver*.

3. Enable USB-debugging on the Android device.
For additional information, see *Enabling USB Debugging*.
4. Ensure that the Open Agent is running on the machine to which the Android device is connected.

When testing a mobile Web application, the Open Agent is automatically used as a proxy for the Android device.



Note: If there is no WiFi connection active when Silk Test tries to hook the mobile Web application, manually set the Open Agent as a proxy for the Android device. For additional information, see *Setting the Open Agent as a Proxy for an Android Device or Emulator*.

5. Open the **Silk Test Web Tunneler** app on the Android device.

Silk Test installs the **Silk Test Web Tunneler** app on the Android device to enable the USB connection between the Open Agent and the device.

6. To test a secure mobile Web application over HTTPS, Silk Test copies a root certificate to the device during hooking. If the certificate is not installed, the Silk Test Web Tunneler app displays a message box, stating that the root certificate is not installed. Click on the message box to install the certificate.



Note: If the certificate is not installed automatically during hooking, see *Manually Adding a Root Certificate to Test a Secure Web Application*.

7. Create a Silk Test project for your mobile Web application.
8. Create a test for your mobile Web application.
9. Use the **Mobile Recording** feature to record the test against the mobile Web application.
10. Replay the test.
11. Analyze the test results.

Testing Mobile Web Applications on an Android Emulator

To test a mobile Web application on an Android emulator, perform the following tasks:

1. Configure the emulator settings for Silk Test.
For additional information, see *Configuring the Android Emulator for Silk Test*.
2. Start the Android emulator.
3. To test a mobile Web application, set the Open Agent as a proxy for the Android emulator.



Note: Ensure that the Open Agent is running on the machine to which the mobile device is connected.

For additional information, see *Manually Setting the Open Agent as a Proxy for an Android Device or Emulator*.

4. To test a secure mobile Web application over HTTPS, Silk Test copies a root certificate to the device during hooking. If the certificate is not installed, the Silk Test Web Tunneler app displays a message box, stating that the root certificate is not installed. Click on the message box to install the certificate.



Note: If the certificate is not installed automatically during hooking, see *Manually Adding a Root Certificate to Test a Secure Web Application*.

5. Create a Silk Test project for your mobile Web application.
6. Create a test for your mobile Web application.
7. Use the **Mobile Recording** feature to record the test against the mobile Web application.
8. Replay the test.
9. Analyze the test results.

Installing a USB Driver

To connect an Android device for the first time to your local machine to test your mobile Web applications, you need to install the appropriate USB driver.

The device manufacturer might provide an EXE with all the necessary drivers for the device. In this case you can just install the EXE on your local machine. If the manufacturer does not provide such an EXE, you can install a single USB driver for the device on the machine.

To install the Android USB driver on Microsoft Windows 7:

1. Find the appropriate driver for your device.
For information on finding and installing a USB driver, see <http://developer.android.com/tools/extras/oem-usb.html>.
2. Connect your Android device to a USB port on your local machine.

3. From your desktop or **Windows Explorer**, right-click **Computer** and select **Manage**.
4. In the left pane, select **Device Manager**.
5. In the right pane, locate and expand **Other device**.
6. Right-click the device name, for example *Nexus S*, and select **Update Driver Software**. The **Hardware Update Wizard** opens.
7. Select **Browse my computer for driver software** and click **Next**.
8. Click **Browse** and locate the USB driver folder.
By default, the Google USB Driver is located in <sdk>\extras\google\usb_driver\.
9. Click **Next** to install the driver.

For information on upgrading an existing USB driver or installing a USB driver on another operating system, see <http://developer.android.com/tools/extras/oem-usb.html>.

Enabling USB-Debugging

To communicate with an Android device over the Android Debug Bridge (adb), enable USB debugging on the device.

1. On the Android device, open the settings.
2. Tap **Developer Settings**.
If the developer settings are not included in the settings menu of the device:
 - a) Depending on whether the device is a phone or a pad, scroll down and tap **About phone** or **About Pad**.
 - b) Scroll down again and tap **Build Number** seven times.
3. In the **Developer settings** window, check **USB-Debugging**.

Manually Setting the Open Agent as a Proxy for an Android Device or Emulator

To set the Open Agent as a proxy for your Android device or Android emulator, install the Open Agent on the machine from which you want to test the device or the emulator and enable USB debugging on the device or the emulator.



Note: When you are testing a real Android device, the Open Agent is automatically set as a proxy for the Android device.

1. On the Android device or Android emulator, open the settings.
2. To use a wireless network connection:
 - a) Click **Wi-Fi**.
 - b) Perform a long click on the active connection.
 - c) Click **Modify network**.
 - d) Check **Show advanced options**.
 - e) Click **Proxy settings**.
 - f) Select **Manual**.
3. To use a 3G connection:
 - a) Click **More**.
 - b) Click **Mobile networks**.
 - c) Click **Access Point Names**.
 - d) Select the active access point.
 - e) Click **Proxy**.

4. Type the IP-address of the machine on which the Open Agent is installed into the **Proxy** or **Proxy hostname** field.
5. If you are configuring a 3G connection, click **OK**. The machine on which the Open Agent is installed is now listed as a proxy for the Android device or the Android emulator.
6. Click **Port**.
7. Type the port for the Open Agent into the **Port** field. By default, the port number is random. To test the AUT on an emulator or to configure a wireless network connection on an Android device, use the configuration setting **ext.http.proxy.port** in the file `AppData\Roaming\Silk\SilkTest\conf\silkproxy.properties` to set a permanent port number. For example, to set the port number to **9999**, set `ext.http.proxy.port=9999`. Then type the port number into the **Port** field.
8. Click **OK**.

The Open Agent is now set as a proxy for your Android device or Android emulator. For additional information on configuring a proxy for your Android device or Android emulator, refer to the documentation of the device or the emulator.



Note: As long as the Open Agent is running, you can use the Internet connection on the mobile device that uses the Open Agent as a proxy. If the Open Agent is not running, the connection will no longer work, and you have to use another connection to connect to the Internet from your mobile device. If you remove the wireless network connection while the device or emulator is still running, the connection to the Open Agent persists until you shut down the device or emulator.

Configuring the Android Emulator for Silk Test

When you want to test mobile Web applications on an Android emulator with Silk Test, you have to configure the emulator for testing:

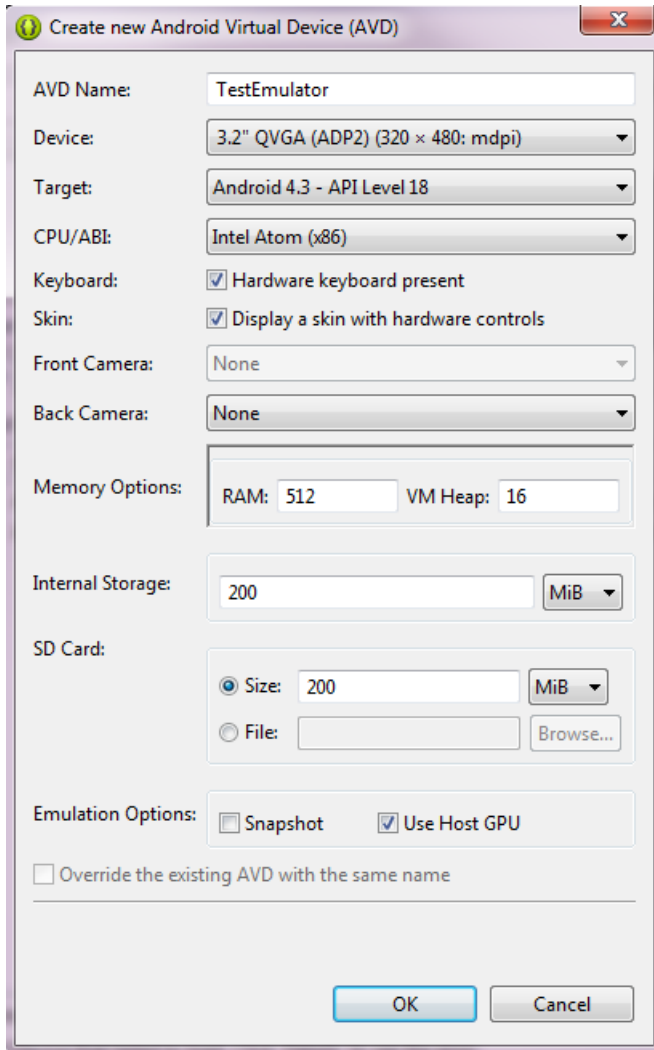
1. Install the Android emulator.
For information on how to install and configure an Android emulator, see [Get the Android SDK](#).
2. From Eclipse, click **Window > Android SDK Manager** to start the **Android SDK Manager**.
3. For all Android versions that you want to test with the emulator, expand the version node and check the check box next to **Intel x86 Atom System Image**.
4. Click **Install** to install the selected packages.
5. Expand the **Extras** node and check the check box next to **Intel x86 Emulator Accelerator (HAXM)**.
6. Click **Install** to install the selected packages.
7. Review the *Intel Corporation license agreement*. If you accept the terms, select **Accept** and click **Install**. The **Android SDK Manager** will download the installer to the `extras` directory, under the main SDK directory. Even though the **Android SDK Manager** says `Installed` it actually means that the Intel HAXM executable was downloaded. You will still need to run the installer from the `extras` directory to get it installed.
8. Extract the installer inside the `extras` directory and follow the installation instructions for your platform.
9. In Eclipse, click **Window > Android Virtual Device Manager** to add a new Android Virtual Device (AVD).
10. Select the **Android Virtual Devices** tab.
11. Click **New**.
12. Configure the virtual device according to your requirements.
13. Set the RAM size used by the emulator to an amount that is manageable by your machine.
For example, set the RAM size for the emulator to 512.
14. Set a size for the SD card.



Note: If you do not set a size for the SD card, you need to set the value of the internal storage to 50 MB or more, otherwise you cannot copy the certificate file to the emulator.


15. To enhance the speed of the transactions on the emulator, select the **Intel Atom (x86)** CPU in the **CPU/ABI** field.
16. *Optional:* To enhance the speed of the transactions on the emulator, you can also check the **Use Host GPU** check box in the emulation options.

 **Note:** In Android, there is an issue with the **Use Host GPU** option. For additional information, see <https://code.google.com/p/android/issues/detail?id=58724>.



17. Click **OK**.

Recording Mobile Web Applications

 **Note:** Some low-level methods and classes are not supported for mobile Web applications. To be able to correctly replay a test recorded against a mobile Web application, uncheck the **Record native user input** option in the Browser options of Silk Test before recording against the mobile Web application. For additional information, see *Limitations for Testing Mobile Web Applications*.

Once you have established the connection between Silk Test and a mobile device or an emulator, you can record the actions that are performed on a mobile browser on the device to create tests. To record mobile Web applications, Silk Test uses the **Mobile Recording** feature, which provides additional functionality compared to the recorder that is used for standard or Web applications.

The **Mobile Recording** feature displays the screen of the mobile device or emulator which you are testing.



Note: If no mobile device is connected to the machine and no emulator is started, the **Mobile Recording** window displays an error message. Connect your mobile device to the machine or start the emulator and then click **Refresh** in the **Mobile Recording** window.

When you perform an action in the **Mobile Recording** feature, the same action is performed on the mobile device.

When you interact with a control on the screen, the **Mobile Recording** feature preselects the default action. A list of all the available actions against the control displays, and you can select the action that you want to perform or simply accept the preselected action by clicking **OK**. You can type values for the parameters of the selected action into the parameter fields. Silk Test automatically validates the parameters.

When you cannot directly interact with a control, for example because other controls are hiding the control, you can click **Toggle Object Hierarchy** in the **Mobile Recording** feature to select the control from the control hierarchy tree.

When you pause the recording, you can perform actions in the screen which are not recorded to bring the device into a state from which you want to continue recording.

When you stop recording, a script is generated with your recorded actions, and you can proceed with replaying the test.

Interacting with a Mobile Device

To interact with a mobile device and to perform an action like a swipe in the application under test:

1. In the **Mobile Recording** window, click **Show Mobile Device Actions**. All the actions that you can perform against the mobile device are listed.
2. Select the action that you want to perform from the list.
3. Continue with the recording of your test.

Troubleshooting when Testing Mobile Web Applications

Why does the Select Application dialog box not display my mobile browsers?

Silk Test might not recognize a mobile device or emulator for one of the following reasons:

Reason	Solution
The mobile device is not connected to the local machine.	Connect the mobile device to the local machine.
The emulator is not running.	Start the emulator.
The Android Debug Bridge (adb) does not recognize the mobile device.	<p>To check if the mobile device is recognized by adb:</p> <ol style="list-style-type: none"> 1. Navigate to C:\Program Files (x86)\SilkTest\ng\agent\plugins\com.microfocus.silktest.adb_15.0.0.6733\bin. 2. Hold Shift and right-click into the File Explorer window. 3. Select Open command window here. 4. In the command window, type <code>adb devices</code> to get a list of all attached devices.

Reason	Solution
	5. If your device is not listed, check if USB-debugging is enabled on the device.
The version of the operating system of the device is not supported by Silk Test.	For information on the supported mobile operating system versions, refer to the Release Notes .
The USB driver for the device is not installed on the local machine.	Install the USB driver for the device on the local machine. For additional information, see Installing a USB Driver .
USB-debugging is not enabled on the device.	Enable USB-debugging on the device. For additional information, see Enabling USB-Debugging .

Why can my mobile device or emulator no longer connect to the Internet?

If you have configured the Open Agent as a proxy for every network connection on your mobile device or emulator, and you are currently not recording or replaying any tests, the mobile device or emulator cannot connect to the Internet. For a physical mobile device you can check the connection status in the **Silk Test Web Tunneler** application.

If the mobile device is connected and the Open Agent is running, and the mobile device still cannot connect to the Internet, check if the proxy settings are correct.

To be able to connect to the Internet when the Open Agent is not running, you can temporarily disable the proxy.

Why does Silk Test search for a URL in Chrome for Android instead of navigating to the URL?

Chrome for Android might in some cases interpret typing an URL into the address bar as a search. As a workaround you can manually add a command to your script to navigate to the URL.

Why can I not record on an Android emulator with Android 4.3?

To record on an Android emulator with Android version 4.3, uncheck the **Use Host GPU** check box in the emulator settings.

Why do mobile applications no longer work when I configure the proxy?

Some mobile applications do not use the global proxy that you can set for the WiFi connection. Browsers and some applications like Gmail use the proxy settings, but most other mobile applications ignore the proxy settings and therefore cannot connect to the Internet while the proxy is set.

What do I do if the adb server does not start correctly?

When the Android Debug Bridge (adb) server starts, it binds to local TCP port 5037 and listens for commands sent from adb clients. All adb clients use port 5037 to communicate with the adb server. The adb server locates emulator and device instances by scanning odd-numbered ports in the range 5555 to 5585, which is the range used by emulators and devices. Adb does not allow changing those ports. If you encounter a problem while starting adb, check if one of the ports in this range is already in use by another program.

For additional information, see <http://developer.android.com/tools/help/adb.html>.

Why do I get the error: Failed to allocate memory: 8?

This error displays if you are trying to start up the emulator and the system cannot allocate enough memory. You can try the following:

1. Lower the RAM size in the memory options of the emulator.

2. Lower the RAM size of Intel HAXM. To lower the RAM size, run the `IntelHaxm.exe` again and choose **change**.
3. Open the **Task Manager** and check if there is enough free memory available. If not, try to free up additional memory by closing a few programs.

Why can I not work with a secure website?

If you cannot test a secure website (HTTPS) on a physical mobile device, try the following:

1. Open the **Silk Test Web Tunneler** application on the mobile device and check if the certificate for the secure website is installed. If the certificate is not installed a warning displays.
2. Click on the warning and select **Ok** to install the certificate.
3. If the certificate is not found on the device, check if the file `root.crt` exists under `sdcard/silk/certs/`.
4. If the file `root.crt` does not exist, copy the file manually by using the **File Explorer**. The certificate might be missing if you have no write permissions on the mobile device.
5. After you have copied the certificate to the device, you can install the certificate by using the **Silk Test Web Tunneler** application or by clicking on the certificate in the file system.

If you cannot test a secure website (HTTPS) on an emulator, manually add the root certificate of the website. For additional information, see *Manually Adding a Root Certificate to Test a Secure Web Application*.

Manually Adding a Root Certificate to Test a Secure Web Application



Note: To perform the steps described in this topic, you must have configured the Open Agent as a proxy for the Android device or Android emulator.

When you are testing a mobile Web application which uses HTTPS on an Android device or Android emulator, each request to open a specific site will automatically generate a certificate for this site on the machine on which the Open Agent is installed. This new certificate is issued to the same domain as the original certificate, replacing the original certificate to enable testing over the SSL connection.

The first certificate that is generated is the root certificate for the mobile Web application.

To be able to test the application with Silk Test, the root certificate needs to be installed on the Android device or Android emulator. By default, the root certificate is copied to the device during hooking. However, if the root certificate is not automatically installed, manually install the root certificate once for each mobile Web application that you want to test.

1. Open the mobile Web application that you want to test. If it is the first time that you open the mobile Web application, the Open Agent generates the modified root certificate for the application.
2. On the machine on which the Open Agent is installed, go to the folder where the root certificate is located.

By default, this is the folder `%Appdata%\Silk\SilkTest\certs\authority`.

3. Copy the root certificate file `root.crt`.
4. Paste the root certificate file to the root folder in the storage of your Android device.

If you are testing on an Android emulator, the Open Agent automatically copies the certificate to the root directory of the emulator.



Note: To enable the Open Agent to copy the certificate to the emulator, configure a size for the SD card in the emulator settings.

5. If you are testing on a physical Android device, install the certificate from the storage into your Android device.

For additional information about how to install a certificate from the storage, refer to the documentation of your Android device or Android emulator.

6. If you are testing on an Android emulator:
 - a) Navigate to **Settings > Security > Install from SD card** on the emulator.
 - b) Click **OK** to install the certificate.
 - c) *Optional:* Navigate to **Settings > Security > Trusted credentials > USER** to verify that the certificate is installed on the emulator.

Limitations for Testing Mobile Web Applications

The support for playing back tests and recording locators on mobile browsers is not as complete as the support for the other supported browsers and native mobile applications. The following list lists the known limitations for playing back tests and recording locators on mobile browsers:

- To click on an object in a mobile Web application, use the `DomClick` method of the `DomElement` class. You cannot use the `Click` method of the `IMobileClickable` interface to click on an object in a mobile Web application.
- The following classes, interfaces, methods, and properties are currently not supported for mobile Web applications:
 - `BrowserApplication` class.
 - `CloseOtherTabs` method
 - `CloseTab` method
 - `ExistsTab` method
 - `GetActiveTab` method
 - `GetSelectedTab` method
 - `GetSelectedTabIndex` method
 - `GetSelectedTabName` method
 - `GetTabCount` method
 - `OpenTab` method
 - `SelectTab` method
 - `DomElement` class.
 - `DomDoubleClick` method
 - `DomMouseMove` method
 - `GetDomAttributeList` method
 - `DomForm` class. All methods and properties in this class are not supported for mobile Web applications.
 - `DomRadioButton` class.
 - `RadioListItemCount` property
 - `RadioListItems` property
 - `RadioListSelectedIndex` property
 - `RadioListSelectedItem` property
 - `DomTable` class. All methods and properties in this class are not supported for mobile Web applications.
 - `DomTableRow` class. All methods and properties in this class are not supported for mobile Web applications.
 - `IClickable` interface.
 - `Click` method

- `DoubleClick` method
- `PressMouse` method
- `ReleaseMouse` method
- `IKeyable` interface. All methods and properties in this interface are not supported for mobile Web applications.
- XPath logical operators are not supported for all attributes. For example, the logical operators are not supported for the `textContent` attribute and the `innerText` attribute.
- Recording in landscape mode is not supported for emulators that include virtual buttons in the system bar. Such emulators do not correctly detect rotation and render the system bar in landscape mode to the right of the screen, instead of the lower part of the screen. However, you can record against such an emulator in portrait mode.

Index

A

- adding root certificates
 - Android 11
- Android
 - configuring emulator 7
 - enabling USB-debugging 6
 - installing USB drivers 5
 - prerequisites 11
 - setting proxy 6
 - testing 4
 - testing on emulators 5
 - testing on physical devices 4
 - troubleshooting 9

D

- device not connected
 - mobile 9

I

- installing USB drivers
 - Android 5

M

- mobile
 - troubleshooting 9
- mobile applications
 - recording 8
- mobile browsers
 - limitations 12

- mobile devices
 - interacting with 9
 - performing actions against 9
- mobile recording
 - about 8
- mobile Web applications
 - limitations 12
 - testing 4
- mobile Web testing
 - Android 4
 - Android emulators 5
 - clients 4
 - overview 4
 - physical Android devices 4

P

- proxy server
 - setting for Android 6

R

- recording
 - mobile applications 8
- root certificates
 - adding 11
 - generating 11

T

- troubleshooting
 - mobile 9