VCL OneClick for iOS

# iOS App

## Features

* Add, delete, edit and display OneClicks
* Create reservations with OneClicks
* Automatic connection to Windows (RDP) and Linux (SSH) reservations
* Support for multiple RDP and SSH clients.
* Display, end, and connect to current reservations
* Universal App (works on both iPhone and iPad)
* Retina display graphics for iPhone and iPad
* Securely store user’s credentials in the iOS keychain with the possibility of clearing the credentials.

## Known Issues

* When creating and editing an OneClick, the platform is not determined, which can cause the connection to the reservation to fail. **Workaround**: refresh list of OneClicks before attempting to connect. **Solution**: add “ostype” field to values returned by getImages() API.
* The “about” screen needs to be formatted properly to accommodate scrolling of the text.

# Deliverables

* iOS App Source Code
* Image resources in both normal and retina resolutions, for both iPhone and iPad, including 4-inch iPhone 5.
* Additions and modifications to VCL’s source code
  + utils.php
  + states.php
  + oneclick.php
  + xmlrpcWrappers.php
* VCL Logo in vector image format (Adobe Illustrator)
* Database scripts
  + Create ‘oneclick’ table
  + Add ‘oneClick’ privilege
* Documentation

# New XMLRPC Methods

**XMLRPCgetIP()**

Returns an array with the following indices:

* status: will always be “success”
* ip: IP address of the requester as seen by the server

**XMLRPCgetOneClicks()**

Returns an array with a list of the user’s OneClicks. The array contains the following indices:

* status: will have a value of “success”, or “error”
* oneclicks: will only be available when the status is “success” and will contain arrays with the following keys:
  + oneclickid: ID of the OneClick
  + name: User-specified name for the OneClick
  + imageid: ID of the image that is assigned to this OneClick
  + imagename: Name of the image that is assigned to this OneClick
  + ostype: The platform of the image. Can be “windows”, “linux” or “unix”
  + duration: The number of minutes to use when creating a reservation
  + autologin: Whether the OneClick should automatically log the user into the reservation once it becomes available (currently not used).

In case of status=”error”, the array will contain the errorcode: errormsg indices with the values described below:

* 1: Unable to retrieve user's OneClicks.
* 5: The user is not allowed to use OneClick functionality.

**XMLRPCgetOneClickParams($oneclickid)**

Returns an array with the details of a OneClick. The array contains the following indices:

* status: will have a value of “success”, or “error”
* name: User-specified name for the OneClick
* imageid: ID of the image that is assigned to this OneClick
* imagename: Name of the image that is assigned to this OneClick
* ostype: The platform of the image. Can be “windows”, “linux” or “unix”
* duration: The number of minutes to use when creating a reservation
* autologin: Whether the OneClick should automatically log the user into the reservation once it becomes available (currently not used).

In case of status=”error”, the array will contain the errorcode: errormsg indices with the values described below:

* 5: The user is not allowed to use OneClick functionality.
* 4: The OneClick with ID $oneclickid does not exist.
* 6: The OneClick with ID $oneclickid does not belong to the user that requested it.

**XMLRPCaddOneClick($name, $imageid, $duration, $autologin)**

Returns an array with the ID of the newly created OneClick. The array contains the following indices:

* status: will have a value of “success”, or “error”
* oneclickid: ID of the OneClick

In case of status=”error”, the array will contain the errorcode: errormsg indices with the values described below:

* 1: Unable to create OneClick
* 5: The user is not allowed to use OneClick functionality.

**XMLRPCeditOneClick($oneclickid, $name, $imageid, $duration, $autologin)**

Returns an array with the new details of the OneClick that was updated. The array contains the following indices:

* status: will have a value of “success”, or “error”
* name: User-specified name for the OneClick
* imageid: ID of the image that is assigned to this OneClick
* imagename: Name of the image that is assigned to this OneClick
* ostype: The platform of the image. Can be “windows”, “linux” or “unix”
* duration: The number of minutes to use when creating a reservation
* autologin: Whether the OneClick should automatically log the user into the reservation once it becomes available (currently not used).

In case of status=”error”, the array will contain the errorcode: errormsg indices with the values described below:

* 1: Unable to update OneClick.
* 5: The user is not allowed to use OneClick functionality.
* 4: The OneClick with ID $oneclickid does not exist.
* 6: The OneClick with ID $oneclickid does not belong to the user that requested it.

**XMLRPCdeleteOneClick($oneclickid)**

Returns an array with the status of the operation. The array contains the following index:

* status: will have a value of “success”, or “error”

In case of status=”error”, the array will contain the errorcode: errormsg indices with the values described below:

* 1: Unable to delete OneClick
* 5: The user is not allowed to use OneClick functionality.

# Updated XMLRPC Methods

**XMLRPCgetRequestIds()**

Updated to return the new item ostype on the details of the requestsReturn. This new item ostype can have the values “windows”, “linux”, or “unix”, and is required to appropriately choose the connection method (RDP or SSH).

# Database Changes

/\* Add new privilege \*/

INSERT INTO userprivtype (name) VALUES ('oneClick');

/\* Create new table and foreign keys \*/

create table oneclick

(

id int not null auto\_increment,

userid mediumint(8) not null,

imageid smallint(5) not null,

name varchar(70) not null,

duration int not null,

autologin tinyint(1) not null default 0,

status tinyint not null default 1,

primary key (id),

INDEX `userid` (`userid`),

INDEX `imageid` (`imageid`)

)

ENGINE=InnoDB;

alter table oneclick add constraint oneclick\_ibfk\_1 foreign key (userid)

references user (id) on delete restrict on update restrict;

alter table oneclick add constraint oneclick\_ibfk\_2 foreign key (imageid)

references image (id) on delete restrict on update restrict;