

# **Xvnc Pro X/X11 Server with VNC Backend**

---

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

## Contents

<b>1</b>	<b>Components</b>	<b>1</b>
<b>2</b>	<b>Installation</b>	<b>1</b>
<b>3</b>	<b>Usage</b>	<b>4</b>
<b>4</b>	<b>User Guide</b>	<b>5</b>
<b>5</b>	<b>Uninstall</b>	<b>6</b>
<b>6</b>	<b>Compatibility</b>	<b>7</b>
6.1	OpenIntents . . . . .	7
6.2	AndroidVNC . . . . .	7
6.3	PocketCloud . . . . .	7
<b>7</b>	<b>FAQ</b>	<b>7</b>
7.1	Xvnc or Xvnc Pro . . . . .	7
7.2	How do I use it? . . . . .	8
7.3	How do I test it? . . . . .	8
7.4	How can I invoke Xvnc Pro from my own X application . . . . .	9
7.5	How can I request the installation from my application . . . . .	9
7.6	AndroidVNC bugs . . . . .	9
7.7	The X server is listening on all interfaces . . . . .	10
<b>8</b>	<b>CREDITS &amp; License</b>	<b>10</b>

---

## Introduction

Xvnc Pro is an implementation of an X Server for Android.

The app is available on Google Play <https://play.google.com/store/apps/details?id=com.theqvd.android.xpro>

Development has been sponsored by Qindel for use with QVD, a leading open source VDI solution. To find out more about QVD, please visit the website: <http://theqvd.com>

---

## 1 Components

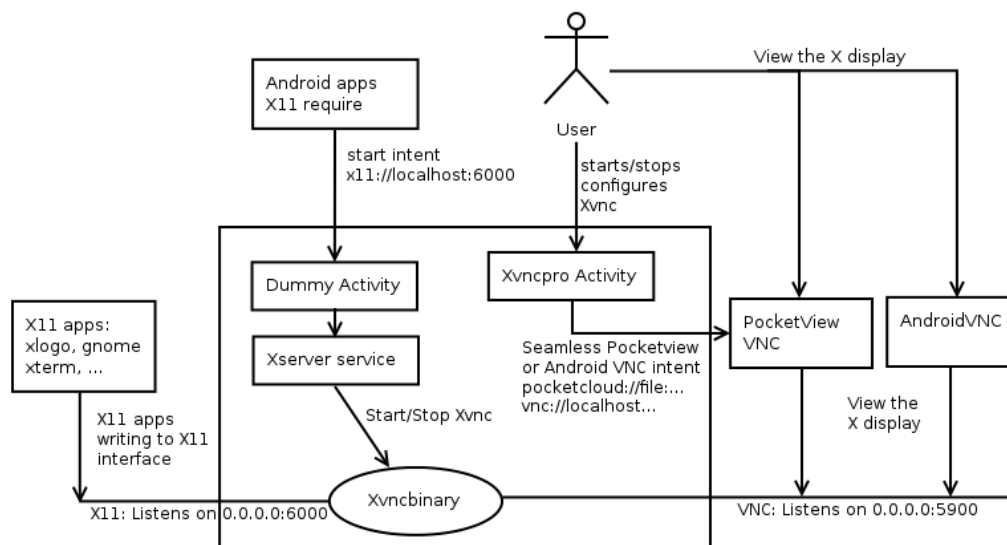
Xvnc Pro consists of an X server that draws in an VNC display instead of drawing directly in the video card. This means that to view what is actually drawn you will need a VNC viewer. The decision to use Xvnc over a native X server stems from the fact that current implementation of X is somewhat immature. Efforts have been made, however, to keep the invocation interface compatible with the android X server.

The list of components is as follows:

- Xvnc Pro application. This consists of the following components:
- Configuration Activity (Screen). Here you can configure and start and stop the X server.
- Dummy Activity (Screen). This is where the fake display is sent.
- Xserver service. This is a background process which has the X server running.
- PocketCloud VNC viewer or AndroidVNC viewer (this last one is included in the app).

External interfaces to this application are:

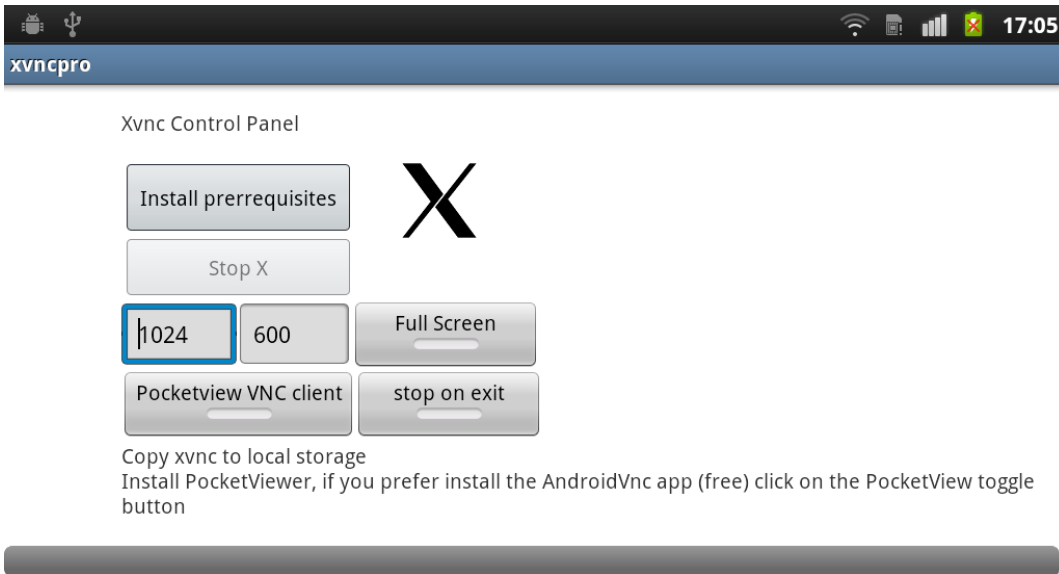
- The X11 interface which listens on TCP/6000 to allow X applications to write in the X Display.
- The VNC interface which listens on TCP/5900 to allow to view the X display via the VNC interface.
- The x11://localhost:6000 interface which allows to launch the X display from another application.



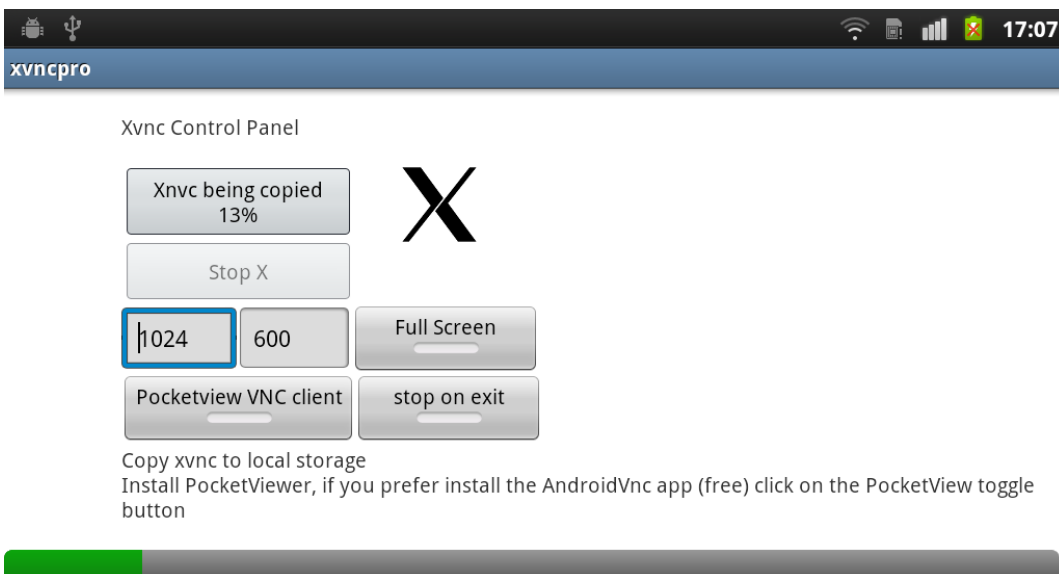
## 2 Installation

Before you can use Xvnc Pro server you will need to perform the following actions:

- Install the app from Google Play and open it.



- Xvnc Pro works in conjunction with a VNC server, the free AndroidVNC or the paid for PocketCloud Remote Desktop Pro, so you will need to select one at this point using the toggle button within the app. For further information on the two servers, please refer to the FAQ section below.
- Select "Install Prerequisites" which will copy the Xvnc binary to the local storage. It will also bring up the Google Play page for the VNC server you have selected, you should proceed to install that.



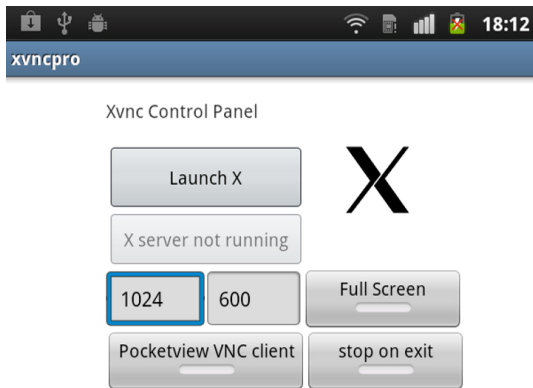
- Install PocketCloud Remote Desktop Pro or,



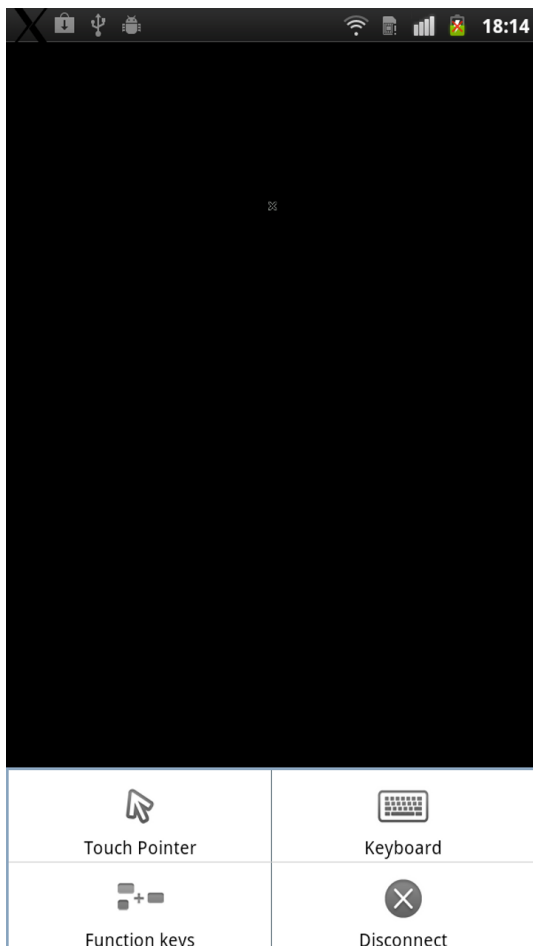
- Use AndroidVNC if you selected that instead (this is included in the application)

### 3 Usage

- Starting and stopping the X server is achieved through the use of a toggle button within the application.



- If you use the PocketCloud VNC server, disconnecting from within there will optionally stop the X server.



- You can also start the X server from your own application by invoking the following Intent (see the FAQ for example code):  
`x11://localhost:6000`

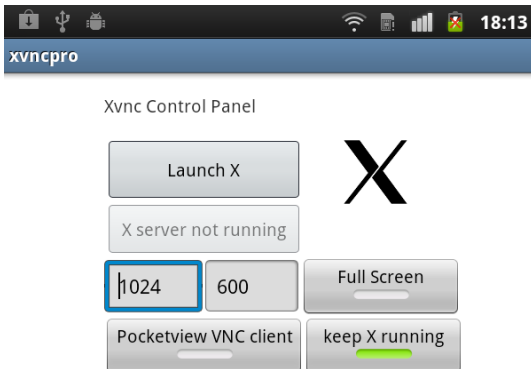


## 4 User Guide

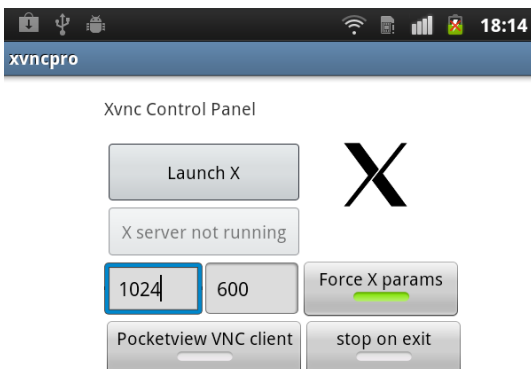
Note: All configuration data is stored in the android property file and is persistent across restarts. That means that you can set up the X resolution or opt to keep the X server running after disconnecting from VNC and Xvnc Pro will remember your settings for next time.

User options:

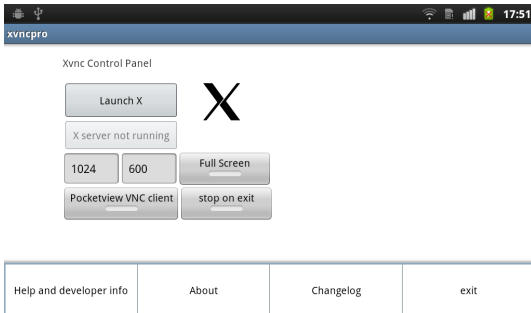
- **Keep running.** Select "keep X running" if you wish to keep the X server running after disconnecting the VNC client. If you use the PocketCloud VNC server, disabling this option stops the X server when your VNC session has completed.



- **Force resolution.** The X server should automatically default to full screen, but if for some reason it doesn't, or should you wish to use another resolution, you may override the default settings here.



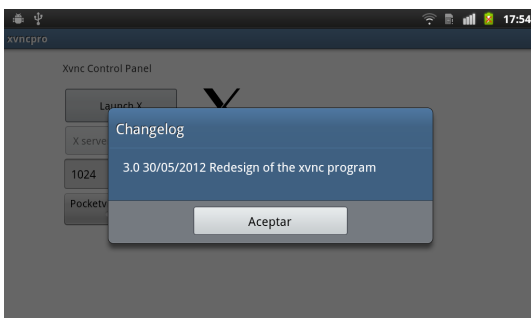
- **Menu options.** Selecting the menu option on your android device will give you an array of help and about information, as well as providing a way to manually exit the application.



- **Help.** This document.
- **About.** Provides licensing information, as well as the version of Xvnc Pro you are using.



- **Changelog.** A list of recent changes to the application.



- **Exit.** Stops the X server and quits the application.

## 5 Uninstall

To uninstall, you will need to go to the Applications > Manage applications in Settings (on devices running Android 4.0 and greater, Settings > Apps), and remove the device through the provided interface.

Currently the binary is packaged inside the application and installed into the following location:

```
/data/data/com.theqvd.android.xpro/
```

The binaries stored in this location might not get deleted on uninstall, so you may wish to delete this folder after uninstalling the app.

If for some reason you want to reinstall the application just delete the following file and restart the application:

```
/data/data/com.theqvd.android.xpro/shared_prefs/PREFS_PRIVATE.xml
```

On upgrades some users have reported that they had to remove the application private data from the application menu.

## 6 Compatibility

### 6.1 OpenIntents

The X11 intent interface `x11://localhost:6000` is defined in `openintents.org`. See

<http://www.openintents.org/action/android-intent-action-view/x11-server>

This intent has been coordinated with the Android X server (a native implementation of X for Android). So you should be able to launch your X applications via the `x11://localhost:6000` intent and be able to choose which X server you want to run. Please see:

<https://play.google.com/store/apps/details?id=au.com.darkside.XServer&hl=en>

### 6.2 AndroidVNC

The AndroidVNC intent used to launch is the following

```
Intent: vnc://localhost:5900/C24bit/benlto
```

By default Xvncpro also responds to `vnc://` intents, so if you want to use it to launch vnc viewers, you can.

### 6.3 PocketCloud

The PocketCloud intent used to launch is the following

```
Intent: pocketcloud://file:///data/data/com.theqvd.android.xpro/files/xvnc.vnc
```

The content of the `xvnc.vnc` file is:

```
full address:s:127.0.0.1
server port:i:5900
pocketcloud password:s:benlto
```

## 7 FAQ

### 7.1 Xvnc or Xvnc Pro

Why should you use Xvnc Pro instead of just Xvnc?

- Xvnc Pro can optionally stop the X server when you disconnect from the session.
- Xvnc Pro supports PocketCloud Remote Desktop Pro (around \$14) which has superior keyboard and mouse to the free AndroidVNC, although the latter is still supported.
- The application interface and installation interface has been reviewed.

When should you use Xvnc instead of Xvnc Pro?

---

- You are happy with AndroidVNC and don't plan to use PocketCloudRemoteVNC.
- You are happy to kill the X server manually or to rely on the Android OS to stop it when needed.
- You don't mind using an non updated application

Why not just use the free AndroidVNC?

We have found several issues, at the time of writing, with AndroidVNC, including:

- Keyboard and mouse issues. Especially with CTRL keys and cursors. You can circumvent these with the Hackers keyboard to certain extent. And there is also a patched AndroidVNC from the Hackers keyboard which does the job. See [http://code.google.com/p/hackerskeyboard/wiki/FrequentlyAskedQuestions#Android\\_VNC\\_Viewer\\_doesn%27t\\_recognize\\_the\\_extra\\_keys](http://code.google.com/p/hackerskeyboard/wiki/FrequentlyAskedQuestions#Android_VNC_Viewer_doesn%27t_recognize_the_extra_keys)
- On VNC disconnection AndroidVNC does not immediately release the connection. See <http://code.google.com/p/android-vnc-viewer/issues/detail?id=299>
- Each vnc intent creates a new connection in the vnc database <http://code.google.com/p/android-vnc-viewer/issues/detail?id=294>

## 7.2 How do I use it?

- Launch the X server.
- This example uses connectbot, but you ought to be able to use and other ssh client that allows port redirection.
- Connect to your desktop with your username and password and then logout.
- Long press on your connection until a menu pops up, and select "Edit port forwards"
- Enter the following details:
- A nickname, eg xvnc
- Type: remote
- Source port: 6010
- Destination: localhost:6000
- Connect again. And from your terminal session type:

```
export DISPLAY=localhost:10
xlogo &
xterm &
icewm &
```

## 7.3 How do I test it?

Here is how we tested Xvnc Pro.

- Launch the X server
- ssh into my android box with port redirect. For that you will ok, so need an ssh server for your Android device, we used SSHDroidPro.
- Run the following command from your terminal:

```
ssh -L 6001:localhost:6000 root@yourandroiddevice
```

- From your desktop computer try the following:

```
export DISPLAY=localhost:1
xlogo &
xterm &
icewm-session &
```

## 7.4 How can I invoke Xvnc Pro from my own X application

You should also be able to launch from your application with the following Intent `x11://localhost:0`

Please be aware that the listening on localhost and on display :0 is currently hardcoded, so if you try something like `x11://localhost:6001` or `x11://10.0.0.1:6000`, the app will still launch `x11://localhost:6000`. If you really have a need to customize these settings, please get in touch and we will consider changing this.

Example code:

```
public class MainActivity extends Activity {
    private Button buttonLaunchX;
    static final String tag = "XtestIntent-MainActivity-" + java.util.Map.Entry.class ←
        .getSimpleName();
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        buttonLaunchX = (Button) findViewById(R.id.button1);
        buttonLaunchX.setOnClickListener(new Button.OnClickListener() {
            public void onClick(View view) {
                String cmd = "x11://localhost:0";
                Log.i(tag, "launching intent:"+cmd);
                Intent x11Intent = new Intent(android.content.Intent.ACTION_VIEW, Uri. ←
                    parse(cmd));
                startService(x11Intent);
            }
        });
    }
}
```

## 7.5 How can I request the installation from my application

Use this code:

```
public boolean installXifNotInstalled() {
    ApplicationInfo info;
    String packagename = "com.theqvda.android.xpro";
    try {
        info = getPackageManager().getApplicationInfo(packagename, 0 );
    } catch( PackageManager.NameNotFoundException e ){
        Log.i(tag, packagename + " is not installed");
        Intent goToMarket = new Intent(Intent.ACTION_VIEW).setData(Uri.parse("market ←
            ://details?id="+packagename));
        startActivity(goToMarket);
        return false;
    }
    Log.i(tag, packagename+" is already installed" + info);
    return true;
}
```

## 7.6 AndroidVNC bugs

- <http://code.google.com/p/android-vnc-viewer/issues/detail?id=299>
- Keyboard

## 7.7 The X server is listening on all interfaces

Currently binding to localhost/127.0.0.1 is not supported. As a workaround during the launch of the Xvnc binary the "-localhost" option is specified disallowing access to the X server from addresses other than localhost.

## 8 CREDITS & License

The Xvnc server is distributed under the GNU Public License Version 2.

The Android Application is distributed under the GNU Public License Version 3.

It was developed by <Nito at Qindel dot ES> and sponsored by the QVD project <http://theqvd.com>

---