**Dimitri**

* Nexus 5 custom ROM
* Settings do disable only some rights for an app

**Response:**

**Q1:** Nexus 5 custom ROM

**A1:** For each different phone we have an unique mode of flashing, for most oft he Nexus devices we have the following steps:

* Copy the file *update.zip* in the root of the SD-CARD
* Reboot in recovery mode using the phone (power+vol up?!)
* Choose *update.zip*
* Flash the device
* Reboot the device

**Once we compile the AOSP what we do?**

*make updatepackage*

*fastboot -w update $ANDROID\_PRODUCT\_OUT/$TARGET\_PRODUCT-img-eng.$USER.zip*

* Official documentation for flashing is located at <http://source.android.com/source/building-devices.html> but a lot depends on the bootloader .
* How to build Android AOSP for Nexus 4 - <http://nosemaj.org/howto-build-android-nexus-4>
* How to build Android 4.3 for Nexus 4 - <http://nosemaj.org/build-android-4-3-nexus-4>
* Nexus 4 si Nexus 5 - <https://developers.google.com/android/nexus/drivers>
* Howto Build Android KitKat (4.4) for the Google Nexus 5 - <http://nosemaj.org/howto-build-android-kitkat-nexus-5>

**Q2:** Settings do disable only some rights for an app

**A2:** My comments goes toward 2 solutions: AOSP App Ops Manager and the Cyangenmod Privacy Guard.

* The *App Ops Manager* code is introduced in the AOSP in the Android 4.3 version and slowly propagates also on the new versions - <https://github.com/android/platform_frameworks_base/blob/master/core/java/android/app/AppOpsManager.java>
* Some screenshots with the screen of the App Ops Manager can be found on <http://goo.gl/9TT50k>
* An interesting history of App Ops Manager can be seen on M. Murphy blog post - <http://goo.gl/JlxVcm>
* As alternative there is also the Cyangenmod Privacy Guard app offering the same functionality on the on an Cayngenmod

**Ivan**

* extract boot images, and create boot images
* boot loader overview for x86
* typical partitions - PAGE XXX
* how Android initialize the network interface, how to handle it from outside (ip, resolver names etc)
* how about the packages folder and how you can use it

**Response:**

**Q1:** Extract boot images, and create boot images

**A1:**

**Q2:** Boot loader overview for x86

**A2:**

**Q3:** Typical partitions - PAGE XXX

**A3:**

**Q4:** How Android initialize the network interface, how to handle it from outside (ip, resolver names etc)

**A4:**

**Q5:** How about the packages folder and how you can use it

**A5:**

**Oleg**

* how to flash images back to phones
* PXE Android ????

**Response:**

**Q1:** how to flash images back to phones

**A1:** Check the response for Dimitri.

**Q2:** PXE Android ????

**A2:**

**Alexander**

* Move notification to the bottom, and slide it from bottom-top
* 180 degree rotation of the screen

**Response:**

**Q1:** Move notification to the bottom, and slide it from bottom-top

**A1:**

**Q2:** 180 degree rotation of the screen

**A2:**

**Pavel**

* Software buttons navigation goes UP!
* Gralloc , hwcomposer - pag XXX
* Power management
* Secure partitions – info from BUC

**Response:**

**Q1:** Software buttons navigation goes UP!

**A1:**

**Q2:** Gralloc , hwcomposer - pag XXX

**A2:**

**Q3:** Power management

**A3:**

**Q4:** Secure partitions – info from BUC

**A4:**

**Alexander**

* Draw 2 pixels on the screen from c++

**Response:**

**Q1:** Draw 2 pixels on the screen from c++

**A1:**

**Alexey**

* complex example of debugging (cross JAVA/C)
* direction Android
* alternative to ART/Dalvik
* overview of network capabilities: what kind of connections are available nfc,bluetooth and maybe future directions

**Response:**

**Q1:** complex example of debugging (cross JAVA/C)

**A1:**

**Q2:** direction Android

**A2:**

**Q3:** alternative to ART/Dalvik

**A3:**

**Q4:** overview of network capabilities: what kind of connections are available nfc,bluetooth and maybe future directions

**A4:**

**Nikita**

* Threading in Android in native/java
* memory model in Android, page size and how to handle it

**Response:**

**Q1:** Threading in Android in native/java

**A1:**

**Q2:** memory model in Android, page size and how to handle it

**A2:**

**Andrey**

* watchdog for native apps

**Response:**

**Q1:** watchdog for native apps

**A1:**

**Sergey**

* ANR ............ (how the system is detecting this)

**Response:**

**Q1:** ANR ............ (how the system is detecting this)

**A1:**

**Andrey**

* Adb hangs on device, how to detect that

**Response:**

**Q1:** Adb hangs on device, how to detect that

**A1:**

**Stepan**

* How simcard is interacting with the phone, they way the phone is „loading“ the SIM apps; it is possibile to „stop“ these apps without breaking calling functionality

**Response:**

**Q1:** How simcard is interacting with the phone, they way the phone is „loading“ the SIM apps; it is possibile to „stop“ these apps without breaking calling functionality

**Q2:**

**Anton**

* Porting game code for multiple platforms

**Response:**

**Q1:** Porting game code for multiple platforms

**Q2:**