

Id : ds-1504

Cross ref : 1499

Result : Unknown

Reason :

Sink statement : Line 590 in the screenshot below

```
577      @Override
578      public void onTextChanged(CharSequence s, int start, int before, int count) {
579          String number = s.toString();
580          if (start == 0 && before == 0 && count == 0) {
581              return;
582          } else if (TextUtils.isEmpty(s) && before > 0) {
583              setCallLogMode();
584              contactsEntryAdapter.resetFilter();
585              list.setSelection(0);
586          } else if (number.equals("#06#")) {
587              showDeviceId();
588          } else if (number.startsWith("#*#") && number.endsWith("#*#")) {
589              String secretCode = new StringBuilder(number).substring(4, number.length()-4);
590              sendBroadcast(new Intent( action: "android.provider.Telephony.SECRET_CODE",
591                                     Uri.parse("android_secret_code://" + secretCode)));
592          } else {
593              setContactsMode();
594              contactsEntryAdapter.getFilter().filter(s);
595          }
596      }
```

Source Statement : Line 206 in the below screenshot

```
204
205      public String formatNumber(String number) {
206          String formattedNumber = PhoneNumberUtils.formatNumber(number, Locale.getDefault().getCountry());
207          return (null != formattedNumber) ? formattedNumber : null;
208      }
209
```

The graph ds-1499_sink.drawio shows the sources of variables passed to the sink statement, the graph comes to point where there is unknown end to predict the exact source of the variable phone number.