SOURCE
DbUtil.Alarm()
label = getString()
Line 53

AlarmNotificationActivity.class
onCreate()
TextView.setText(label)
Line 60

label

The same TextView is accessed
ahead to append the label

AlarmNotificationActivity.java
onIntent()
t.getText() + ", " + label

label

AlarmNotificationActivity.class
onNewIntent()
TextView.setText(label);

label

AlarmNotificationService.class
handleTriggerAlarm
labels += ", " + label;
Line 215

labels

AlarmNotificationService.class
handleTriggerAlarm
Notification.setContentText(labels)
Line 227

label

AlarmNotificationService.class
refreshNotifyBar()
next_label = DbUtil.Alarm(Cursor).label
Line 362

next_label

AlarmNotificationService.java
refreshNotifyBar()
new Notification.Builder(this)
.setContentTitle(...? ... : next_label);
Line 370

label

AlarmOptions()
OptionsView()
setEdit(..., alarm.label, ...);

alarm.label

AlarmOptions()
OptionsView()
method setEdit()
EditText.setText(label);

The EditText with id R.id.setting_edit is not accessed anywhere except in setEdit()

The variable label generated in the source, ultimately get written inside a TextView, or an Edit Text or gets appended to another string.
In no data flow, does the source variable reach the sink.

Thus the dataflow is classified as FALSE POSITIVE