Archived:Querying Keypad status using GetIndicatorPayload() does not work with custom screensaver (Known Issue)

Archived: This article is **archived** because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

The article is believed to be still valid for the original topic scope.

Overview

`MScreensaverPluginHost::GetIndicatorPayload(TScreensaverIndicatorIndex aIndex, TIndicatorPayload & aResult);` with `EScreensaverIndicatorIndexKeyGuardState` as the first parameter always returns a non-zero value irrespective of the keypad status.

Description

To create a customized screensaver, `MScreensaverPluginHost::GetIndicatorPayload()` needs to be used to query the status of Keypad, missed calls, new messages, active profile, etc. Using this API for querying the keypad status does not work in any of the S60 2nd Edition, S60 3rd Edition, and S60 3rd Edition, FP1 devices.

How to reproduce

```
TIndicatorPayload result;
TInt status = MScreensaverPluginHost::GetIndicatorPayload (
                    EScreensaverIndicatorIndexKeyGuardState, result );

// If result is other than '0', keylock is activated
if (status == KErrNone && result.iInteger != 0)
{
     // Draw key symbol
}
```

Solution

A workaround is to query the keylock status with the `RAknKeyLock` method.

```
RAknKeyLock keyLock;
if (keyLock.Connect() == KErrNone)
    {
    TBool locked = keyLock.IsKeyLockEnabled();
    if (locked)
        {
        // Draw key symbol
        }
    keyLock.Close();
    }
```