How to automatically terminate a thin-client session that has been idle for a specific period of time.

It is often necessary and desirable to prevent the misuse of user licenses or to avoid idle thin-client threads on the isCOBOL Application server. Automatically terminating a thin-client session that has been left idle on an accept statement of a screen for an extended period can help achieve this.

This functionality is currently available by leveraging some of the configuration properties of the isCOBOL runtime framework.

The first property to configure is iscobol.accept_timeout. For example, setting it to 120 will cause all accept statements in the application to automatically terminate if no input is provided after 2 minutes (120 seconds).

However, accept statements often use loop logic to remain active until the user presses an exit button or key (such as Escape). Therefore it is necessary to create a special program to handle the automatic termination of the accept statement. This program will use the iscobol.hot_key.progname=99 property which specifies that when the termination value is 99 (triggered by the automatic timeout), the program identified by 'progname' will be executed.

In the MYSTOP program, you can include logic to execute a STOP RUN sentence, which will automatically terminate the entire session.

Here's an example:

client.properties

```
#to automatically close the application after 2 minutes of inactivity
iscobol.accept_timeout=120
iscobol.gui.accept.before_time.repeat=1
iscobol.hot_key.mystop=99
##
```

The client.properties will be used by the thin-client session with the -c option.

```
iscclient -port 10999 -hostname xxx.yyy.zzz.aaa  -c client.properties PROG
```

The PROG isCOBOL program might look like the following sample:

```
PROGRAM-ID. prog.
WORKING-STORAGE SECTION.
77  ks is special-names crt status pic 9(5).
```

```
SCREEN SECTION.
01  S1.
    03 ENTRY-FIELD LINE 2 COL 2 SIZE 10.
    03 ENTRY-FIELD LINE 4 COL 2 SIZE 10.
    03 PUSH-BUTTON LINE 20 COL 2 SIZE 10
      TITLE "MESSAGE" EXCEPTION-VALUE 1.
    03 PUSH-BUTTON LINE 20 COL 20 SIZE 10
      TITLE "END" EXCEPTION-VALUE 27.
PROCEDURE DIVISION.
MAIN.
    display standard window.
    display S1
    perform until ks = 27
      accept S1 on exception continue end-accept
      if ks = 1
        display message "F1"
      end-if
    end-perform
    goback.
```

And the MYSTOP program would have this logic:

```
PROGRAM-ID. mystop.
WORKING-STORAGE SECTION.
copy "isgui.def".
77  w-num     pic 9 value 0.
77  w-resp    pic 9.
PROCEDURE DIVISION.
MAIN.
    if w-num = 0
      display message
      "If this session remains inactive for other 2 minutes, your "
      "session will be terminated automatically !!!" x"0a"
      "Press No, to avoid it!"
       type mb-yes-no
       giving w-resp
      if w-resp = mb-no
        move 0 to w-num
      else
        move 1 to w-num
      end-if
    else
      stop run
    end-if.
    goback.
```

To test, run PROG in thin-client mode and leave the screen unattended for 2 minutes. You will see the message box from the MYSTOP program appear. If you press the Yes button and then leave the screen unattended for another 2 minutes, the session will terminate, executing the STOP RUN command in the MYSTOP program. You can design the logic of the stop program as needed; for instance, it could terminate the session immediately without warning if desired.

Online URL: https://support.veryant.com/phpkb/article.php?id=348