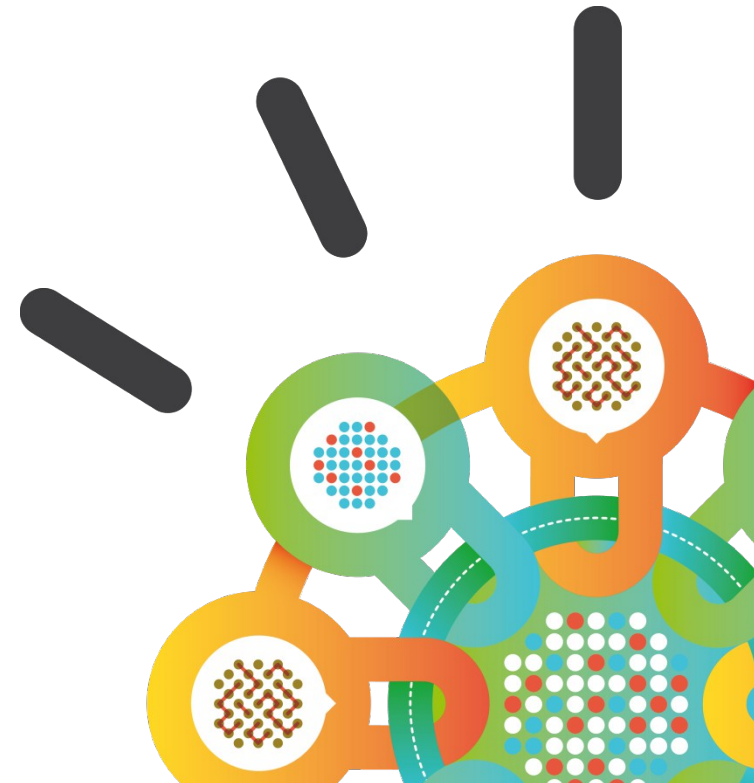


Security Intelligence.
Think Integrated.

IBM Security Access Manager for Enterprise Single Sign-On IBM Rational Host On-Demand (HOD) Profiling Guide





Prerequisites

- **ISAM ESSO AccessAgent 8.2.1 SSE** is installed in any HOD client machines.
- **IBM HOD EHLLAPI Bridge** is installed in any HOD client machines.
 - ISAM ESSO uses EHLLAPI to enable single sign-on (SSO) for HOD green screens (3270 Display, 5270 Display, and VT Display).
- **“Auto-start HLLAPI Enabler”** on configured session for green screens is set to **“Yes”**.
 - On the configured session's **Properties**, select **Preferences > Start Options** to set this option.

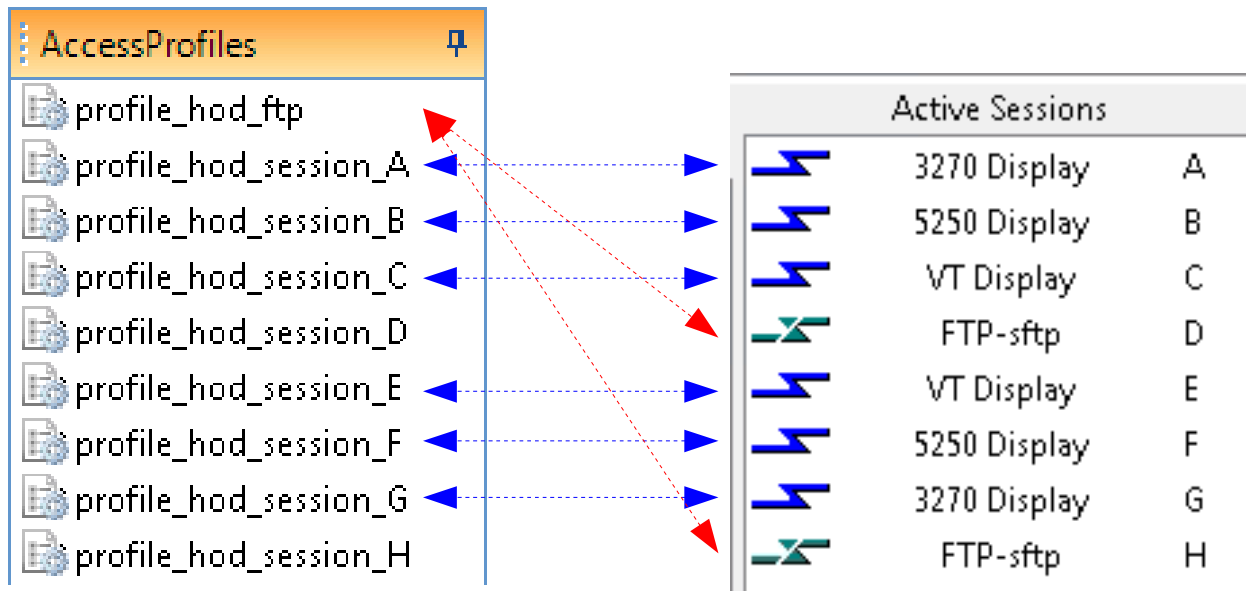
The screenshot shows the 'Start Options' dialog box. It contains the following fields and options:

- Session ID:** A dropdown menu with 'A' selected.
- Start Automatically:** Radio buttons for 'Yes' and 'No', with 'No' selected.
- Start in Separate Window:** Radio buttons for 'Yes' and 'No', with 'Yes' selected.
- Auto-Start Options:** A section containing:
 - Applet/Macro:** A dropdown menu with 'None' selected.
 - Name:** An empty text field.
 - Parameter (Optional):** An empty text field.
- Auto-start HLLAPI Enabler:** Radio buttons for 'Yes' and 'No', with 'Yes' selected. This section is highlighted with a red rectangle.

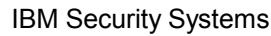


AccessProfile Solution Overview

- Focus : green screen sessions (3270, 5270, and VT Display) and FTP sessions.
- Solution : **concurrent AccessProfiles** for HOD multi-sessions.
 - 1 – 26 AccessProfile(s) for HOD Session ID [A-Z] → for all session types except FTP.
 - 1 AccessProfile for FTP → FTP Login window is modal, so at any time there can only be 1 FTP Login window regardless of how many FTP sessions are launched.



HOD Max. Active Session: 26



AccessProfile for HOD session ID [A-Z] – Part 1

Example: profile_hod_session_A, profile_hod_session_B, ..., profile_hod_session_Z.

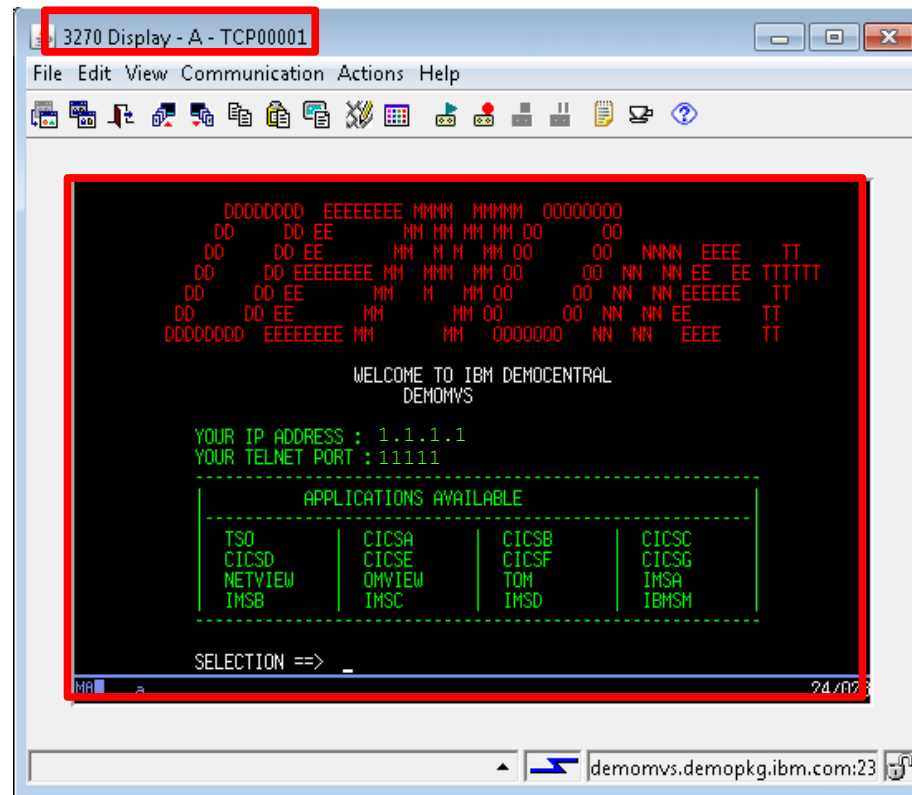
- Mainly targeted for green screen sessions.

Window Title:

contains the session ID, e.g. A. Use this session ID to differentiate any window signatures for trigger/action in the profile.

Java Controls:

located outside the Terminal Display. Use Java-related triggers/actions for these controls.



Terminal Display:

is the green screen emulation. To single sign-on to this space, use HLLAPI and keyboard input simulation.

Example of 3270 Display session



AccessProfile for HOD session ID [A-Z] – Part 2

- There are 3 different areas of triggers and actions to be used.
 - **HLLAPI** (*must be enabled*): for monitoring text changes on the green screen area.
 - **Windows control Keyboard input** (*enabled by default*): for injection/capture.
 - **Java control** (*must be enabled*): for buttons/fields outside the green screen area.
- The signature where the AccessProfile is to be loaded **MUST** be “**java.exe**”.

* Signatures identifying web-page or exe where this AccessProfile is to be loaded

/child:exe[@exe_name="java.exe"]

Add...

Important: because the HLLAPI communication for HOD happens in java.exe process.

- ISAM ESSO supports IBM 32-bit Enhanced EHLLAPI for HLLAPI communication, so **pcshll32.dll** should be referred to in the AccessProfile General Properties.

☒ Enable HLLAPI support for mainframe applications

Dll relative path

C:\Program Files (x86)\IBM\EHLLAPI\pcshll32.dll

☒ Enable support for Java applications



AccessProfile for HOD session ID [A-Z] – Part 3

- You must consider setting the HOD client Session ID for green screens to a **static ID** (A, B, ..., or Z) for the following reasons:
 - The number of AccessProfiles that are created and maintained is the same as the number of the static Session ID used.
- Example:** 3 HOD sessions are configured to use Session ID A, B, and C respectively, so only *profile_hod_session_A*, *profile_hod_session_B*, and *profile_hod_session_C* are created.
- Each AccessProfile can be designed specifically for the session configured with the respective static ID. (1 SSO workflow per AccessProfile)

Start Options

Session ID: A

Start Automatically: ☐ Yes ☒ No

Start in Separate Window: ☒ Yes ☐ No

Auto-Start Options

Applet/Macro: None

Name:

Parameter (Optional):

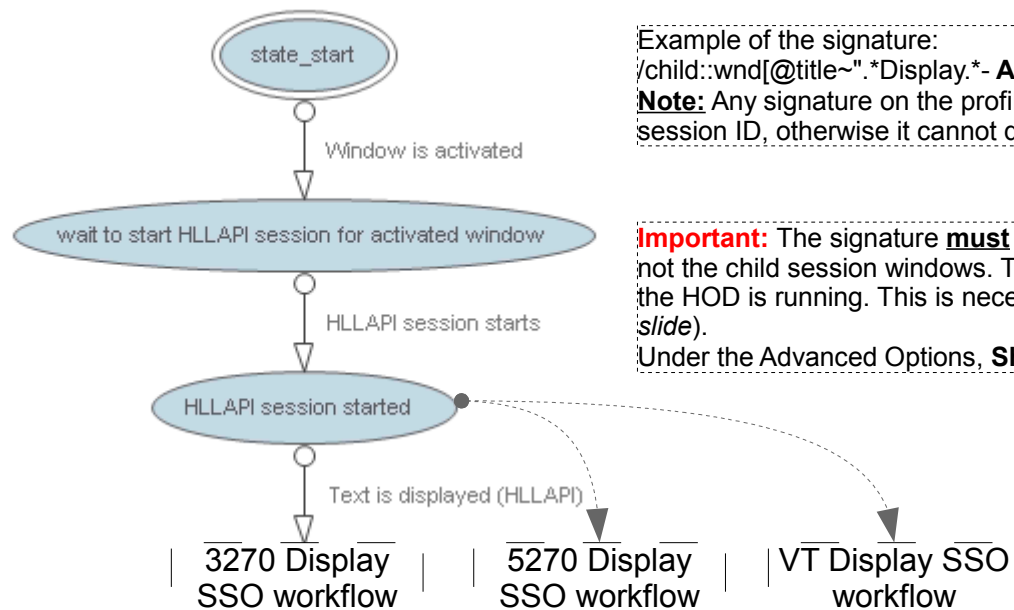
Auto-start HLLAPI Enabler: ☒ Yes ☐ No

Example of SessionID is set to A



AccessProfile for HOD session ID [A-Z] – Part 4

- In situation where the HOD client Session ID needs to be set to **Automatic**, you must take note of the following:
 - The number of AccessProfiles that are created and maintained is 26. This is the maximum number of active HOD sessions, unless the HOD end-users can be enforced to only open up to a certain number of active HOD sessions.
 - Each AccessProfile should handle all the different possible session types used. If the display differs for each session type or each connected host, you need to create more than 1 SSO workflow per AccessProfile.



Example of the signature:

```
/child::wnd[@title~".*Display.*- A$"].*Display.*- A -.*" and @class_name="SunAwtFrame"]
```

Note: Any signature on the profile, whenever appropriate, must have the title denoting session ID, otherwise it cannot distinguish the control in the presence of multiple sessions.

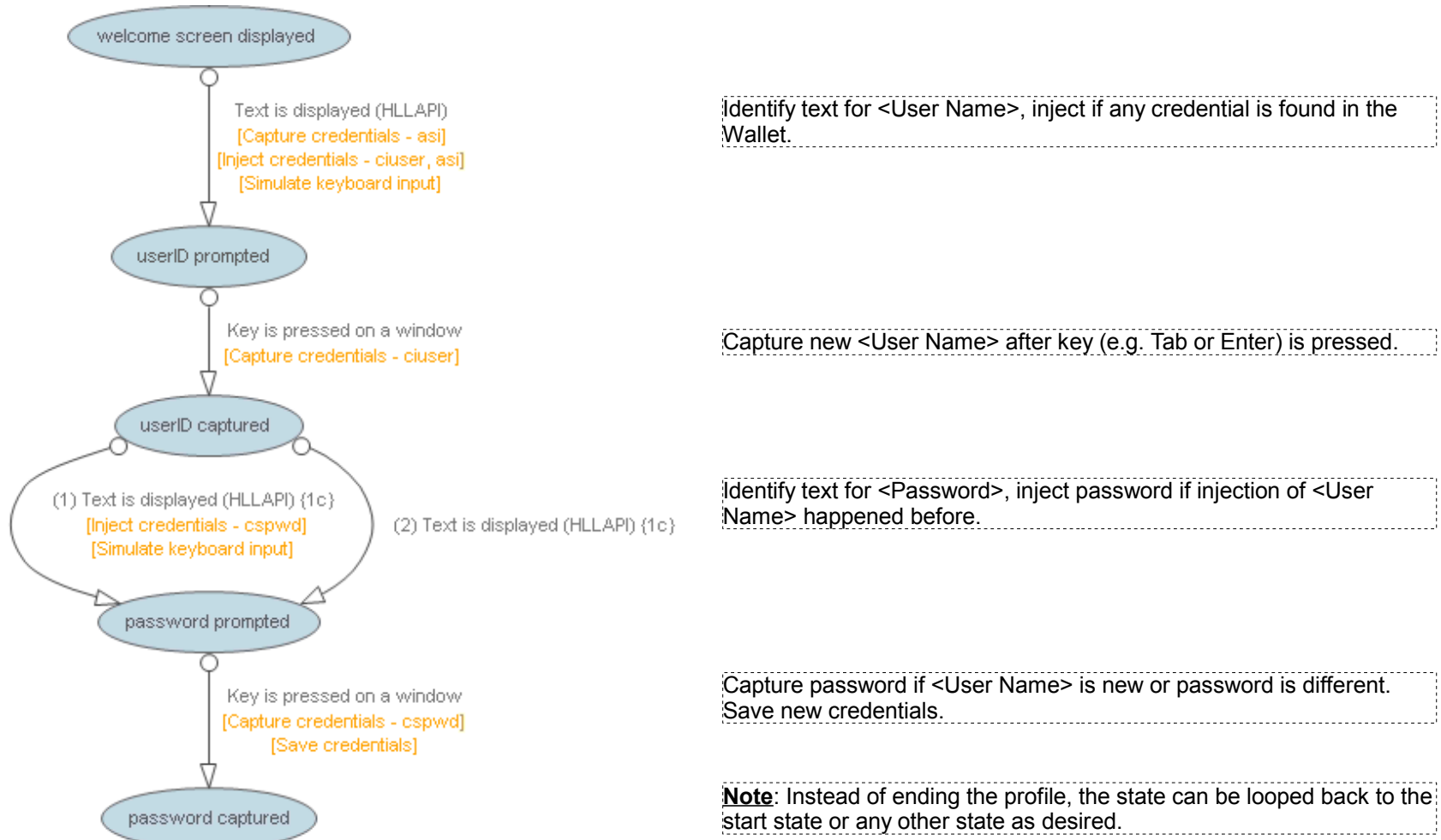
Important: The signature **must** use the **HOD main window (embedded in the browser)**, not the child session windows. The referenced window must always be available as long as the HOD is running. This is necessary for HLLAPI. (Refer to the "Troubleshooting Tips - #2" slide).

Under the Advanced Options, **Short name** should be set to **A**.



AccessProfile for HOD session ID [A-Z] – Part 5

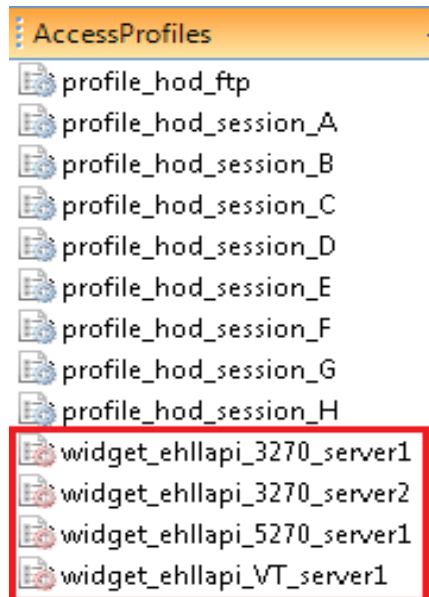
- The green screen SSO workflow is similar to how any mainframe application is profiled for SSO, except that for HOD, HLLAPI trigger is used for the text matching.





AccessProfile for HOD session ID [A-Z] – Optimization: Widget – 1

- Since there are many different SSO workflows to be profiled and these workflows might be repeated for all or some of the concurrent HOD session ID [A-Z] → best candidate to use AccessProfile Widget!
- Benefits of using AccessProfile Widget:
 - Modularity → Easier AccessProfile maintenance. Any changes or updates on a workflow can be performed in one place.
 - Memory saving → Widget instances are not loaded in the memory until the pinned entry state is reached.



Several widgets can be created for **different session types** and **different connected server** as necessary when the SSO workflows differ. The appropriate widgets should be referenced by the main *profile_hod_session_[A-Z]* accordingly.

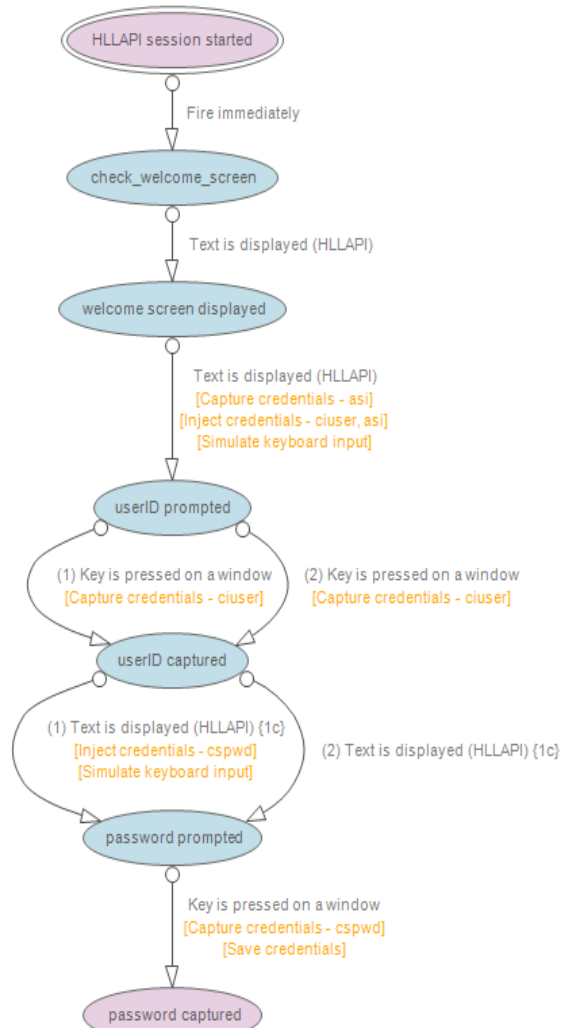
Note: The widgets on the left side are for illustration purposes only.

For example, you can create *widget_ehllapi_3270_demomvs* to profile the 3270 Display SSO workflow to *demomvs* server.



AccessProfile for HOD session ID [A-Z] – Optimization: Widget – 2

- Example of AccessProfile Widget *widget_ehllapi_3270_demomvs* for 3270 Display SSO workflow.



Widget Entry Point

The SSO workflow is similar to any other mainframe application profile.

Any window signature that is used needs to be generic so that the widget can be referenced from any other main AccessProfiles and still identify the respective window correctly.

Note: The signature of a particular control might be the same for all session windows. What differentiates the control is the session window title that contains the Session ID. Ensure that the Session ID is added in the window signature.

Use the **??PROPERTY_NAME** in any window signature field to get the value of this PROPERTY_NAME as the window signature.

Example:

Property **WND_SIGNATURE** is created in the main *profile_hod_session_A* with **Use local bag = Yes**, having the value of `/child::wnd[@title~".*Display.*- A.*" and @class_name="SunAwtFrame"]`

The property can then be referred in any window signature field. To use this property in a widget, pass it to the widget instance as a parameter.

* Signature of the window getting the key-down msg

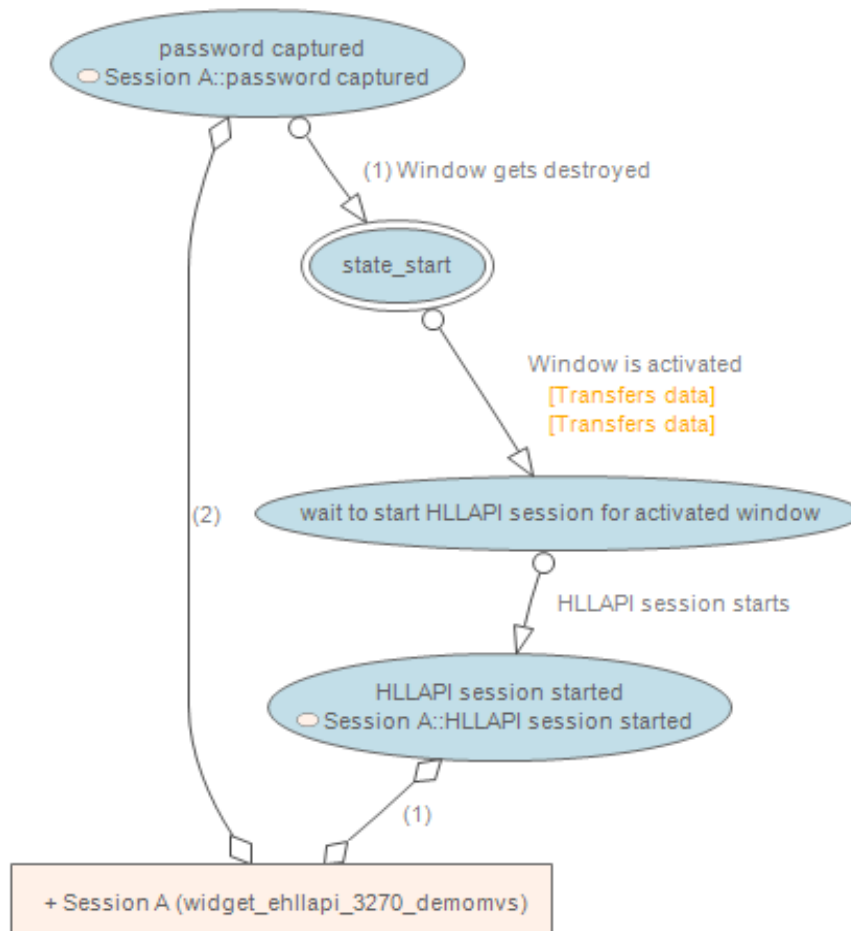
??WND_SIGNATURE

Widget Exit Point



AccessProfile for HOD session ID [A-Z] – Optimization: Widget – 3

- Example of the main AccessProfile *profile_hod_session_A* referencing *widget_ehllapi_3270_demomvs*.



Once the expected window is identified, create a Property to store the unique window signature containing the Session ID. The Session ID is then passed into the widget instance, and referred in the widget instance.

Example:

WND_SIGNATURE = /child::wnd[@title~".*Display.*- A.*" and @class_name="SunAwtFrame"]

Use local bag = Yes --> so that only the widget instance (in this case, is named "Session A") in the main profile can use this property value.

Important: The signature **must** use the **HOD main window (embedded in the browser)**, not the child session windows. The referenced window must always be available as long as the HOD is running. This is necessary for HLLAPI. (Refer to the "Troubleshooting Tips - #2" slide).

Under the Advanced Options, **Short name** should be set to **A**.

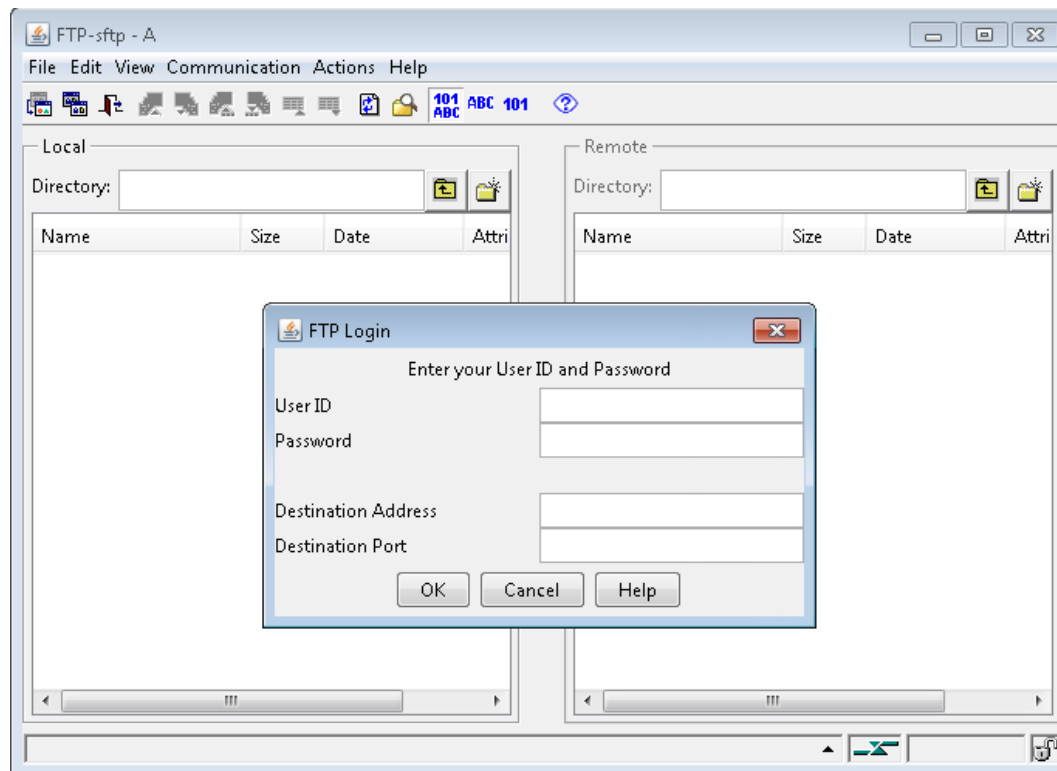
The main profile, *profile_hod_session_A*, is uses the widget *widget_ehllapi_3270_demomvs*. The name of this widget instance for this profile is "Session A".



AccessProfile for FTP – Part 1

Example: profile_hod_ftp.

- Create AccessProfile like a common Java Applet application.
- **Java control** related triggers and actions are used. (*must be enabled*)

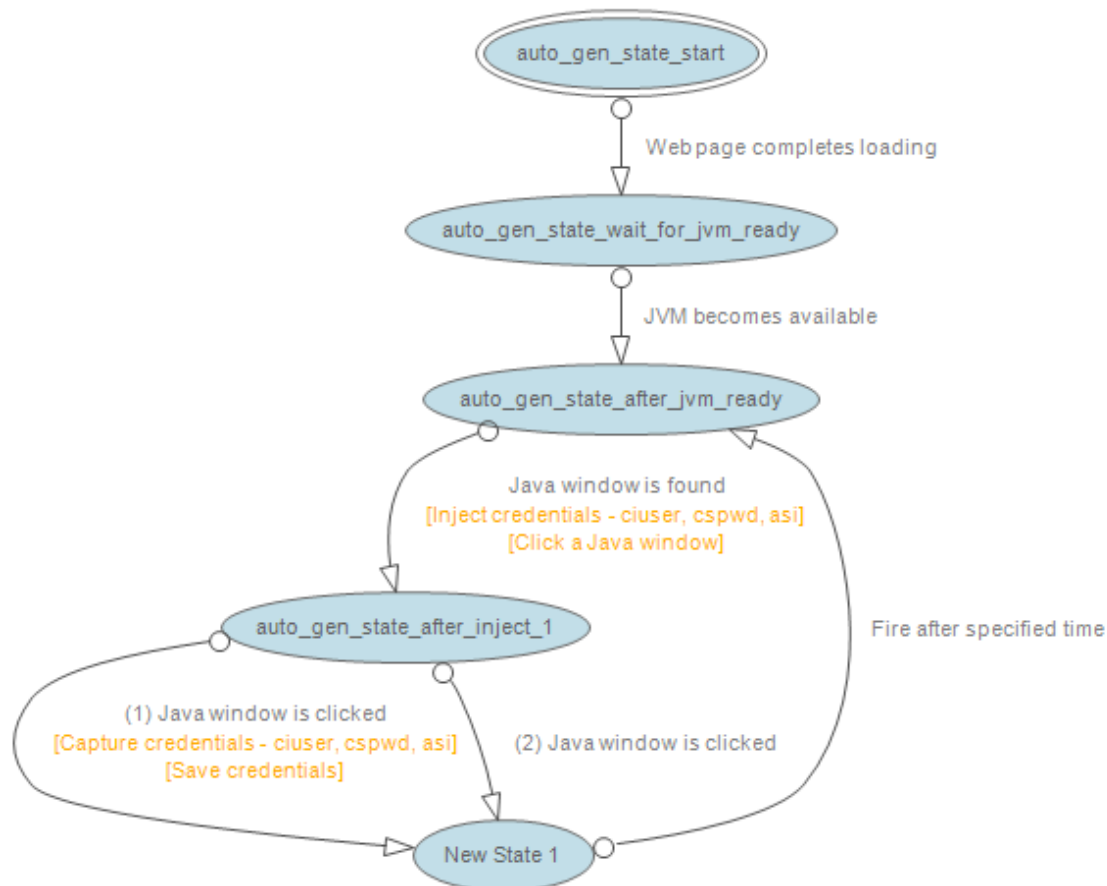


Example of FTP session



AccessProfile for FTP – Part 2

- Example of the AccessProfile *profile_hod_ftp*, similar to SSO to any other Java Applet applications. This profile can be created using AccessStudio Generator.





Checklist of Required AccessProfile Settings

Settings	profile_hod_session[A-Z]	profile_hod_ftp
Profile Signature	/child::exe[@exe_name="java.exe"]	Any normal applet signature
"Can be loaded concurrently"	Yes	Yes
"Enable support for Java applications" Note: Ensure that Java Support for ISAM ESSO is enabled. If not, please run JVMSupport.vbs under ECSS\JavaSupport\ folder.	Checked	Checked
"Enable HLLAPI support for mainframe applications"	Checked (pcshll32.dll to be referred)	N/A



Troubleshooting Tips – #1

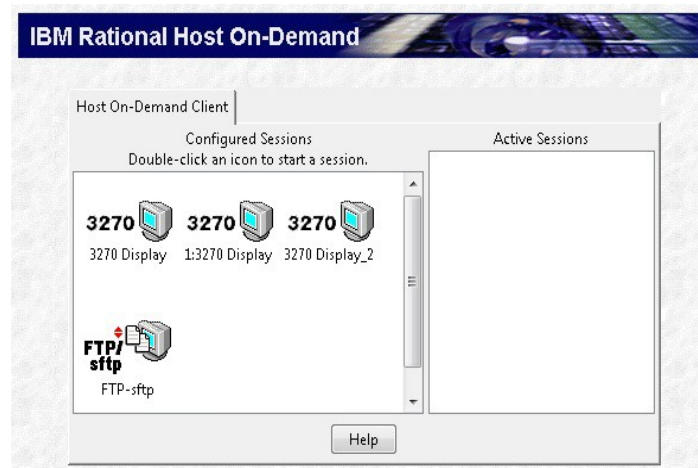
- “**HLLAPI session starts**” trigger is not fired
 - Check the configuration of HOD client and ISAM ESSO to enable HLLAPI support. Ensure that all the **Prerequisites** (*Refer to page 2*) and the **Required AccessProfile Settings** (*Refer to page 14*) are met.



Troubleshooting Tips – #2

- HOD session window is closed, but CPU usage is high and/or the closed session window still appears in the list of Active Sessions on the HOD Main Window.
 - Check that the signature that is used for “**HLLAPI session starts**” trigger is using the **HOD Main Window**, e.g.:

```
/child::wnd[@title~".* - Windows Internet Explorer" and  
@class_name="IEFrame"]/child::wnd[@class_name="Frame  
Tab"]/child::wnd[@class_name="TabWindowClass"]/child::wnd[@class_name="Shell DocObject  
View"]/child::wnd[@class_name="Internet Explorer_Server"]/child::wnd[@class_name="Java Plug-in  
Control Window"]/child::wnd[@class_name="SunAwtFrame"]/child::wnd[@class_name="SunAwtCanvas"]
```
 - If incorrect signature is used, there might be dangling ObsHllProvider.exe (ISAM ESSO process to support HLLAPI) after the session window is closed. This will result in high CPU usage. To recover from this, kill the dangling ObsHllProvider.exe.

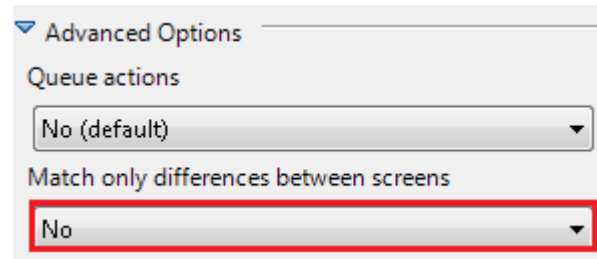


HOD Main Window (embedded in the browser)



Troubleshooting Tips – #3

- “**Text is displayed (HLLAPI)**” trigger is not fired on the first screen display (for example: on the welcome page), even though the text to be matched is present.
 - On the first screen display, check that the option “**Match only differences between screens**” for “Text is displayed (HLLAPI)” trigger is set to “**No**” to search for the entire screen, otherwise it will only check the successive differences which in this case there is none.



- Check on the correctness of the “**Match text (regex)**” that is used to match the text on the screen.



Troubleshooting Tips – #4

- Some of the Actions that require window signature or Java window signature are not executed for *profile_hod_session_[A-Z]* (HOD HLLAPI-related profile)
 - Since *profile_hod_session_[A-Z]* (HOD HLLAPI-related profile) can contain both the normal window triggers/actions and the Java-related triggers/actions, the mixed usage of both types might cause unexpected behaviours.
 - When an Action requires any window signature, check that its parent trigger are of the same type, i.e. if the Action requires any normal window signature, check that its parent trigger is a normal window trigger; if the Action requires any Java window signature, check that its parent trigger is a Java-related trigger.
 - If an Action requires > 1 window signature (for example: inject/capture action), one needs normal window signature (for example: for keyboard input) and the other one needs Java window signature (for example: the text of a field outside the green screen area), they should be split.

Example: An *asi* requires an Indirect Auth Service that is obtained from the Status Bar text of the session window which is a **Java Control**, but to capture the user name for *ciuser*, **Windows control (keyboard input)** must be used.

(1) Key is pressed on a window
[Capture credentials - ciuser, asi]

OR

(1) Key is pressed on a Java window
[Capture credentials - ciuser, asi]

Only *ciuser* will be captured.

Only *asi* will be captured.

Instead, split them

HLLAPI session starts
[Capture credentials - asi]

(1) Key is pressed on a window
[Capture credentials - ciuser]

Note: Any Action whether or not it requires a normal window or Java window signature under HLLAPI-related triggers will work as per normal.



Other Useful References

- **ISAM ESSO HLLAPI Documentation**

http://www.ibm.com/support/knowledgecenter/SS9JLE_8.2.1/com.ibm.itamesso.doc_8.2.1/AccessStudio_Guide/concepts/AS_Advanced_AP_Custom_Requirements_Legacy_Support_HLLAPI.html

- **ISAM ESSO AccessProfile Widget Documentation**

http://www.ibm.com/support/knowledgecenter/SS9JLE_8.2.1/com.ibm.itamesso.doc_8.2.1/AccessProfileWidgets_Guide/concepts/ic_aswidget_guide.html

Security Intelligence.
Think Integrated.

The End

