

Network Station Manager Version 2



Kiosk Mode
(Full Screen Solutions)



Network Station Education
IBM Network Computer Division
June 1999

Objectives



- **What is Kiosk mode (fullscreen solutions)**
- **Login modes**
 - Review Normal (non-kiosk) mode
 - Suppressed login panel mode
 - True Kiosk mode
- **Login Process flow**
- **Kiosk Templates**
 - Location
 - Sample
- **Configuring suppressed mode**
- **Security considerations**

Notes



The topic of this presentation is using the kiosk mode of the Network Station.

The objective of this section is to provide an overview of what kiosk mode (sometimes called full screen mode or full screen solution) consists of, the different ways that kiosk mode can be set up, and how to configure to use the Network Station in kiosk mode.

What is Kiosk Mode?



- **Simulates a real 3270 screen (Only a Power On button)**
- **Suppresses any user interaction**
- **Two methods**
 - True kiosk mode
 - Suppressed login panel mode
- **Kiosk templates provided for:**
 - 3270, 5250, VTxxx Emulators
 - ICA (Windows Applications)
 - UNIX Common Desktop Environment
 - Netscape Navigator
 - Java Application and applets (appletviewer)
- **Why use Kiosk mode?**
 - Lobby units
 - Single application uses and CRT replacement



Kiosk mode means using the Network Station in such a way that it simulates as closely as possible the behavior of a real 3270 non-intelligent terminal where all the user needs to do is to flip the power on button in order to start using an application.

The key objective of kiosk mode is to eliminate any user interaction between the time that the terminal is powered and the time that the application (whatever that application is) is available to the user. Mainly, that means that we do not want the user to have to enter a user name and password, and we want to present the user with a single application (instead of a desktop).

There are two main ways that kiosk mode can be implemented:

- True kiosk mode, which is new with V2R1
- Suppressed login mode, which was used in the previous release (V1R3)

We will explain the differences between these two modes in a moment.

When kiosk mode is implemented for a specific station, it is set up so that one specific application is automatically started when the station is powered on. This application can be one of the following, and there are templates that are provided for each of these:

- An emulator (3270, 5250 or VTxxx)
- An ICA session (Connection to a Metaframe server)
- A Browser (Netscape 4.5)
- A Unix Common Desktop Environment session
- A Java application or Java applet

There are probably two main reasons for using kiosk mode:

- Either we want to put a Network Station in a public place, such as a building lobby, where anybody can use it and where we therefore want to restrict the use of the terminal to a single purpose application
- Or, the station is destined for a user that only requires the use of a single specific application at all times and that user does not need any other application. We therefore simplify the user interface by eliminating anything other than the application he needs.

Normal (non-kiosk) Processing



- The kernel reads the system and terminal configuration profiles
- The login processing is initiated by the kernel
- The login process displays a login panel to the user
- The user enters a user ID and password
- Login validates the user ID and password with an authentication server
- User configuration profiles are downloaded and processed
- The window manager is started
- The user's desktop is displayed
- The application is started

IBM Network Station Login

Username: bechard

Password: *

Server name nsedv2r1 at (9.24.104.218)

Ok Start Over Roam Help



To better understand how kiosk mode works, let's first take a brief look at how a normal non-kiosk mode environment is started.

After the station is powered on, the boot processing takes place resulting in the operating system being downloaded. The operating system then reads the terminal configuration profiles after which it initiates login processing.

A login panel is displayed so that the user can enter a user name and password; this data is validated with an authentication server and user specific configuration files are then downloaded and processed, the window manager is launched and the window manager displays a desktop which is customized for that user.

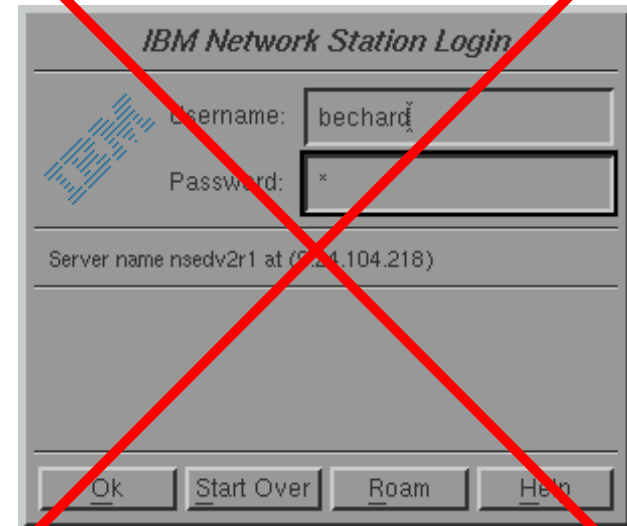
At that point, one or more applications can be automatically started or the user can manually start the application or applications that he or she wants to work with.

The key point is that the login processing is dependent on and based on the identification of a specific user.

Suppresses Login Panel - Kiosk Mode



- The kernel reads the system and terminal configuration profiles
- The login processing is initiated by the kernel
- The login process ~~displays a login panel to the user~~ gets the user name and password from a special file
- ~~The user enters a user ID and password~~
- Login validates the user ID and password with an authentication server
- User configuration profiles are downloaded and processed
- The window manager is started
- The user's desktop is displayed
- The application is started in full screen





Here are listed similar processing steps than those we described in the previous chart but this time, we have crossed out the steps that are not performed when we use the Suppressed Login Panel method of the kiosk mode. As mentioned before, this is the method that was used in the previous release of NSM.

After the kernel gets downloaded, it initiates the login processing as if this was a non-kiosk mode, but notice that instead of displaying a login panel to the user, the login process suppresses the display of the login panel and fetches a user name and password from a special file in which were prerecorded a user name and password, based on the terminal identification. This is where the name Suppressed Login Panel comes from.

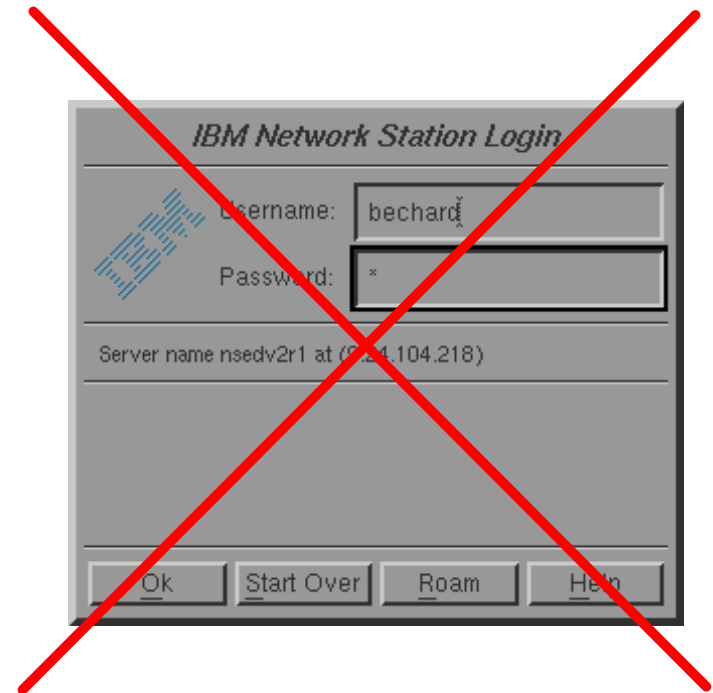
Now that the login process has a user name and password, the remainder of the login process is the same as for a non-kiosk mode. One difference though here is that more than one application could be started if required by configuring more than one application in the startup folder.

In summary, the main difference here is that the user does not see a login panel but we nevertheless used a normal login processing by simply getting the data from a file containing prerecorded data.

True Kiosk Mode



- The kernel reads the system and terminal configuration profiles
- The login processing is initiated by the kernel
- ~~● The login process displays a login panel to the user gets the user name and password from a special file~~
- ~~● The user enters a user ID and password~~
- ~~● Login validates the user ID and password with an authentication server~~
- ~~● User configuration profiles are downloaded and processed~~
- The window manager is started
- ~~● The user's desktop is displayed~~
- The application is started





Now let us take a look at the real kiosk mode that can be used in V2R1.

As can be seen from this list, we have now eliminated all the steps that were part of the login process.

In other words, not only have we eliminated the user interaction but we have in fact eliminated the entire login processing itself.

This is actually accomplished by using a special configuration profile that identifies this station as a kiosk mode station and triggers the kernel to process this special configuration file without requiring the identity of a user.

Summary - True Kiosk vs. Suppressed Login



Summary	True Kiosk Mode	Suppressed Login Panel Mode
User authentication	No	Yes
Access to server file system	Read-only	Read/write
Behaviors	System	System, group, and user
Window manager	Yes, but initial application runs in borderless mode	Yes
Desktop	No desktop. The application is the base window with no borders	Configurable
Applications	One only, but it may launch others or open other windows	One or more
Application launching	Auto-started Auto-restarted	Configurable



So, what is the difference again between these two kiosk mode?

Maybe this table can give you a better idea as to the differences between these two modes. In true kiosk mode:

- There is no specific user identity, whereas there is a specific user in suppressed login mode, even though the user himself did not enter a name, because the name was fetched from a file.
- Access to the server must be read-only for all functions. Since there is no user identity, there is no need to store user preferences for example.
- Since there is no user identity, there is only system and group behavior that's applicable. (default group since there is no real user)
- Actually, the window manager is active but it is not apparent because the application runs full screen without any borders
- There is no desktop in true mode
- There is only one application running (although some applications can actually launch other applications)
- The application is autostarted in both cases, but more than one application can be launched when in suppressed login panel mode.
- In true kiosk mode, the application can be restarted if ended by the user (but not in suppressed login panel mode).

Use Kiosk or Suppressed Login?



- **Some kiosk limitations:**

- Requires more memory because files normally written to the user's home directory are written in memory
- Startup time slightly longer because has to initialize a home directory every time it starts
- User's home directory cannot be primed with any data so you always get default application objects such as a default set of bookmarks, default security certificates, etc.
- Even though configuration changes can be made, they are not remembered across sessions because the home directory gets re-created every session

- **Use Suppressed Login when:**

- Want compatibility with previous release
- Need to give the user more than one application
- Need to remove some of the above limitations
- Need a real user account that can be configured through NSM
- A user always logs into the same physical station and the station is physically secure (screen lock does not secure the desktop since a simple reboot gets it back)



Generally, it is better to use true kiosk mode because it is more secure and more simple.

However, true kiosk mode has some minor limitations:

- It requires a bit more memory because the files normally written to the user's home directory are actually written in the IFS (in-memory file system, like a RAM disk)
- The browser startup time may be slightly longer because it has to recreate and reinitialize the user's home every time
- The user's home directory cannot be primed with any data so you always get default application objects such as a default set of bookmarks, default security certificates, etc.
- Since the user's home directory is always empty to start with, configuration changes cannot be remembered across sessions. So, even though configuration changes can be made, they are only remembered until the end of the session.

So when should one use Suppressed Login mode?

- When you need compatibility with the previous release
- When you need to give the user more than one application
- When you need to remove some of the limitations above



Let us take a closer look now at the login process flows. After the login process is initiated by the kernel, the following decisions take place:

- If a NSM_KIOSK_MODE environment variable exists and is set to ON (this would have set in the special terminal configuration profile), the login simply bypasses the rest of the login process, as shown in (1), mounts the filesystem as read-only, launches the window manager (always present) and launches the specific application that was identified in the special kiosk profile (we will see an example of a profile in a moment).
- If there was no NSM_KIOSK_MODE environment variable, and no kiosks.nsl file exists, we are not in any kiosk mode and the normal login processing takes place from that point on, as shown in (2), that is, a login panel is displayed to the user, the user name and password entered are validated with an authentication server, and the user initialization (based in the user identity) is performed, the window manager started and the desktop manager is launched resulting in the display of a desktop specific to that user. The user then starts one or more applications, or the applications in the startup folder are autostarted.
- On the other hand, if the kiosks.nsl file does exist, and the IP address of this station exists in the file, the user name and password are retrieved from the kiosks.nsl file as shown in (3), and the remainder of the login processing is similar to the normal non-kiosk mode, where we validate the user name that was retrieved from the kiosks.nsl, as shown in (4) and launch the user initialization, launch the window manager and the desktop, and autostart one or more applications.
- In (6), upon logout from the desktop, a cleanup is performed (unmounts, clearing user registry info, etc.) and the login process is restarted.
- In (7), upon logout from a true kiosk application, login simply restarts the window manager and the application since the environment has not changed.

Kiosk Templates Location



The screenshot displays the file structure of the NetworkStationV2 directory. The main window shows the following structure:

- Desktop
 - My Computer
 - 3½ Floppy (A:)
 - (C:)
 - jdk1.1.8
 - NetworkStationV2
 - Patches
 - prodbase
 - servbase
 - bin
 - cgi-bin
 - configs
 - defaults
 - html
 - Locale
 - nls

The right pane of the main window lists the following files:

- allusers.nsm
- appletviewer.ksk
- ica.ksk
- icaram.ksk
- java.ksk
- netscape.ksk
- ns3270.ksk
- ns5250.ksk
- nsmdef.kyr
- nsterm.ksk
- preflist.nsm
- regschem.nsm
- shipped.nsm
- unix.ksk

Callouts from the right pane point to the following labels:

- Java applet
- Single ICA session
- ICA chooser
- Java application
- Netscape browser
- 3270
- 5250
- VTxxx
- Unix CDE

An inset window shows the 'NetworkStationV2' directory with the following structure:

- NetworkStationV2
 - Patches
 - prodbase
 - servbase
 - uninstall
 - userbase
 - home
 - nsmshared
 - profiles
 - groups
 - ncs
 - users

The right pane of the inset window shows the file '000629670213.nsm'.

Notes



This chart shows the location of the supplied kiosk templates (...servbase/defaults) where each particular template bears the name of a ****.ksk file.

If a station needs to be operated in kiosk mode, the administrator must copy one of these templates to the ../userbase/profiles/ncs directory and rename the file to either:

- mac_address.nsm
- ip_address.nsm
- ip_host_name.nsm

where mac_address is the MAC address of the station to be operated in kiosk mode (etc.).

The example shown here uses 000629670213.nsm. It could have been 9.24.105.189.nsm or ethnct1.itso.ral.ibm.com.nsm. If there are multiple files, the system looks first for the IP_address.nsm, then the MAC_address.nsm and then IP_hostname.nsm.

Once a template file has been copied and renamed, the administrator can modify the template to suit his purposes.

Sample Template (3270 Kiosk)



```
<?xml version="1.0" encoding="UTF-8"?>
<!-- 3270 Emulator Sample Profile for Single Application (kiosk) Mode -->
<NCREGISTRY>
<OBJECT NAME="/config">
<CATEGORY NAME="WORKSTATION">
<PROPERTY NAME="pref-screen-background-color">black</PROPERTY>
</CATEGORY>
</OBJECT>
<OBJECT NAME="/desktop/preferences">
<CATEGORY NAME="DESKTOP">
<PROPERTY NAME="desktop_command">nsm_wrapper ns3270 -geometry 9999x9999+0+0</PROPERTY>
</CATEGORY>
</OBJECT>
<OBJECT NAME="/login/session">
<CATEGORY NAME="KIOSK">
<PROPERTY NAME="commands" TYPE="LIST" ACTION="APPEND">
<ELEMENT>
<FIELD NAME="op">SET</FIELD>
<FIELD NAME="arg1">NSM_KIOSK_MODE</FIELD>
<FIELD NAME="arg2">ON</FIELD>
</ELEMENT>
</PROPERTY>
</CATEGORY>
</OBJECT>
</NCREGISTRY>
```



This chart illustrates one of the sample template profiles, in this case the 3270 emulator template (ns3270.ksk).

Notice in particular the XML statement at the end (in blue on this chart) where the NSM_KIOSK_MODE environment variable is set to the value ON to indicate that the station is to operate in kiosk mode.

Also notice the statement `<PROPERTY name="desktop_command"` where the actual application to be started is identified, in this case the ns3270 application.

In this example, only the general 3270 interface is started and the user must enter the address of the host when prompted, but the administrator could modify the template to specify a specific host to be contacted, and also to add any other parameter that is allowed by the ns3270 command.

Suppressed Login Panel Configuration



- On the server, create a user ID to be used as a kiosk user ID
- Add this user ID to the NSMUser group
- Configure an application to be autostarted
 - Configure the application in the launchbar and add the application to the startup folder of the launchbar

- Create a kiosks.source file

IP Host	User ID	Password
7.24.104.189	kiosk3270A	passworda
7.24.104.190	kiosk3270B	passwordb
X.X.X.X	User ID	password

- Encode the kiosks.source file

- The nsmkiosk.exe utility is located in /servbase/bin/
 - (Note: On AIX, the utility is called createKIOSK)
- It takes kiosks.source file (can be any name) and encodes the data
- It stores the file as kiosks.nsl file into /userbase/profiles/ncs



To use the Suppressed Login Panel mode instead of true kiosk mode, the following configuration steps need to be performed by the administrator:

- On the authentication server, create a user name (and password) that will be used as the kiosk user name to be placed in the kiosks.nsl file
- Make this user name part of the NSMUser group, just like any other user
- For this user, use the NSM launchbar configuration to define the application that needs to be autostarted and place that application in the Startup folder to make it autostart
 - Note: That application would normally be started in full screen mode. However, it is possible to start more than one application, in windowed or fullscreen mode. The last one started is simply the one that is in the foreground.
 - Note: If all kiosk users are similar, you can define a kiosk group instead, or start multiple terminals with the same user name (however, beware of the default settings and preferences)
- Create a kiosks.source file (the name can be anything you want) and for each station that needs to operate in suppressed login mode, enter the IP address of the station, and the user name and password to be used by that station.
- Use the nsmkiosk utility to encode the file into the kiosks.nsl file, which is placed by the utility in the ../userbase/profiles/ncs directory. The presence of that file triggers the login function to read the file to see if this particular station is to be operated in suppressed login mode.

Suppressed Login and the Launchbar



- **If you keep launchbar, make sure to disable the lock function**
 - The user does not have the password to unlock the station afterwards
- **If you do not want a launchbar, configure a default launchbar with nothing in it but the startup folder**
 - Applications are auto-started from the startup folder
 - Applications can be closed by the user and restarted from the startup folder
 - If the user logs out, the desktop and applications are automatically restarted
- **The launchbar can also be completely removed by using an override file and setting the Registry to auto-start an application instead of the desktop**
 - You are then nearly in true kiosk mode but not quite since you are still logging on as a specific kiosk user



In suppressed login mode, the user gets an application autostarted without having entered any data whatsoever but having only done a power on. However, a desktop still comes up and the default launchbar is displayed, which permits the user to use any of the application icons located on the launchbar. There is nothing wrong with this, dependent on what is on the launchbar that you may or may not want a user to have access to.

In all cases, you should first ensure that the launchbar for a kiosk user does not have the lock function enabled because the user does not know the password that was used and he will not be able to unlock the station once it is locked (will need to power off/on).

If you still want to keep a launchbar but have a minimal amount of applications available from the launchbar, you can use NSM to configure the system default launchbar with very few items or no item at all (the startup folder is always there) and then customize groups (a kiosk group for example) or user with their individual launchbars.

There is also a way to suppress entirely the desktop and the launchbar by using an override file to set some registry entries that will cause an application to start instead of the desktop manager.

Security Considerations



- The kiosks.source file contains **unencoded passwords** so it should be adequately protected from access
- Encoding program (nsmkiosk.exe) should be accessible only by the system administrator
- Kiosk user IDs should have limited authority
- The kiosks.nsl file should be writable only by the system administrator
- If the file system used cannot prevent a general user from creating a kiosks.nsl, an empty file should be created and protected by the system administrator
- The unencoded file should not be kept in any directory that clients have read access to



When using the suppressed login panel, there are certain security considerations that must be taken into account.

- Because the kiosks.nsl file contains unencoded passwords, it should be adequately protected against unauthorized access and the encoding program (nsmkiosk.exe) should only be accessible by the administrator.
- The user IDs used as kiosk IDs should have limited authority and the kiosks.nsl file should be writable only by the administrator
- Finally, if the file system used cannot prevent a general user from creating a kiosks.nsl, an empty file should be created and protected by the system administrator, and the unencoded file should not be kept in any directory that clients have read access to

Where to Go for More Information



- **Main Web Site**

- www.ibm.com/nc

- **Current Network Station Redbook**

- **SG24-5844 Network Station Manager V2R1 Guide**

- **Previous Network Station Redbooks**

- **SG24-5187 AS/400 - Techniques for Deployment in a WAN**

- **SG24-5221 Windows NT - NSM Release 3**

- **SG24-5212 Printing**

- **SG24-2127 Windows NT/WinCenter**

- **SG24-4954 S/390, SG24-2016 RS/6000, SG24-2153 AS/400**

- **Product Publications**

- **SC41-0684 Installing NSM for AS/400**

- **SC41-0685 Installing NSM for RS/6000**

- **SC41-0688 Installing NSM for Windows NT**

- **SC41-0690 Using NSM**

- **IBM Network Station Advanced Information (On the Web Site)**

Notes



Use this Web site, redbooks and product publications for more information.