Platform LSF
Version 9 Release 1.3

Using on Windows

SC27-5311-03

Platform LSF
Version 9 Release 1.3

Using on Windows

SC27-5311-03

Note
Before using this information and the product it supports, read the information in "Notices" on page 33.

First edition

This edition applies to version 9, release 1 of IBM Platform LSF (product number 5725G82) and to all subsequent releases and modifications until otherwise indicated in new editions.

Significant changes or additions to the text and illustrations are indicated by a vertical line (|) to the left of the change.

If you find an error in any Platform Computing documentation, or you have a suggestion for improving it, please let us know.

In the IBM Knowledge Center, add your comments and feedback to any topic.

You can also send your suggestions, comments and questions to the following email address:

pccdoc@ca.ibm.com

Be sure include the publication title and order number, and, if applicable, the specific location of the information about which you have comments (for example, a page number or a browser URL). When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright IBM Corporation 1992, 2014.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Chapter 1. Test Your IBM Platform LSF Installation 1
Check the cluster 1
Check the Load Information Manager (LIM) 2
Check the Remote Execution Server (RES) 2
LSF on EGO 2
Check LSF 4

Chapter 2. IBM Platform LSF Default User Mapping 7
About LSF default user mapping 7
Specify user names 9
Configure LSF default user mapping 10
Syntax substitution for Windows user names . . . 11

Chapter 3. Environment 13
Job execution environment 13
Control the execution environment using job starters 14

Chapter 4. Charting Resources with Windows Performance Monitor 15
LSF Monitor statistics 15
Install LSF Monitor 16
Configure LSF Monitor 16
Administer LSF Monitor 17

Chapter 5. Dynamic IP Addressing for LSF Hosts 19
How LSF works with dynamic IP addressing . . . 19
Set up DHCP clients 19

Chapter 6. Displaying a Graphical User Interface in LSF with Microsoft Terminal Services 21
How LSF works with Remote Desktop Services . . 21
Requirements 23
Configure Remote Desktop Services for LSF . . . 23
Configure Windows Server 2008 for TS jobs . . . 24
Configure Windows Server 2012 for TS jobs . . . 25
Configure LSF to run Remote Desktop Services jobs 25
Use Windows Remote Desktop Services Job Support on Windows client version host 26
Submit LSF jobs to Terminal Services hosts (tssub) 26
Limit the number of Remote Desktop Services jobs on a host 27
Submit a Remote Desktop Services job from UNIX 27

Chapter 7. Installing LSF in a Mixed Cluster. 29
Set up a Linux cluster with Windows compute nodes 29
Set up a Windows cluster with Linux compute nodes 31
Set up a Windows cluster with Linux compute nodes and EGO controlling LSF daemons 32

Notices 33
Trademarks 35
Privacy policy considerations 35

© Copyright IBM Corp. 1992, 2014

Chapter 1. Test Your IBM Platform LSF Installation

Before you make LSF available to users, make sure LSF is installed and operating correctly. You should:

- Check the cluster configuration
- Start the LSF daemons (LSF services)
- Verify that your new cluster is operating correctly

If you have a mixed UNIX and Windows cluster, make sure you can perform operations from both UNIX and Windows hosts.

Check the cluster

Before using any LSF commands, wait a few minutes for LSF services to start.

1. Log on to any host in the cluster.
2. Check the configuration files.

   ```
   C:\LSF_9.1.3> lsadmin ckconfig -v
   ```

   Typical output is as follows:

   ```
   C:\LSF_9.1.3>lsadmin ckconfig -v
   Checking configuration files ...
   Platform EGO 1.2.6, Nov 2 2012
   binary type: nt-x86
   Reading configuration from C:\LSF_9.1.3\conf\ego\cluster1\kernel/ego.conf
   Dec 21 08:38:59 2012 4196:1492 6 9.1.3 Lim starting...
   Dec 21 08:38:59 2012 4196:1492 6 9.1.3 LIM is running in advanced workload execution mode.
   Dec 21 08:38:59 2012 4196:1492 6 9.1.3 Master LIM is not running in
   EGO_DISABLE_UNRESOLVABLE_HOST mode.
   Dec 21 08:38:59 2012 4196:1492 5 9.1.3 C:\LSF_9.1.3\9.1.3\etc/lim.exe -C
   Dec 21 08:38:59 2012 4196:1492 7 9.1.3 setMyClusterName: searching cluster files...
   Dec 21 08:38:59 2012 4196:1492 7 9.1.3 setMyClusterName: local host hostA belongs to

   Dec 21 08:38:59 2012 4196:1492 3 9.1.3 domanager(): C:\LSF_9.1.3\conf/lsf.cluster
   .cluster1(13): The cluster manager is the invoker <LSF\lsfadmin> in debug mode
   Dec 21 08:38:59 2012 4196:1492 6 9.1.3 reCheckClass: numhosts 1 so reset exchIntvl to 15.00
   Dec 21 08:38:59 2012 4196:1492 7 9.1.3 getDesktopWindow: no Desktop time window configured
   Dec 21 08:38:59 2012 4196:1492 6 9.1.3 Checking Done.
   ---------------------------------------------------------
   No errors found.
   ```

3. Start the LSF cluster.
   a. If you have a Windows-only cluster, start the LSF cluster:

      ```
      C:\lsf\9.1.3\bin> lsfstartup
      ```

      This command starts the LSF services, LIM, RES, and SBD on all LSF Windows hosts. It could take up to 20 seconds.

   b. If you have a mixed UNIX-Windows cluster, you need to log on to a UNIX host and start the UNIX daemons with `lsfstartup`, and then log on to a Windows host and use `lsfstartup` from a Windows host to start LSF services on all Windows hosts.
4. Display the cluster name and master host name:

   ```
   lsid
   ```

© Copyright IBM Corp. 1992, 2014

Check the Load Information Manager (LIM)

1. Display cluster configuration information about resources, host types, and host models:

   `lsinfo`

   The information displayed by **lsinfo** is configured in `LSF_CONFDIR\lsf.shared`.
2. Display configuration information and status of LSF hosts:

   `lshosts`

   The output contains one line for each host in the cluster. Type, model, and resource information is configured in the `LSF_CONFDIR\lsf.cluster.`*cluster_name* file. The cpuf matches the CPU factor given for the host model in `LSF_CONFDIR\lsf.shared`.
3. Display the current load levels of the cluster:

   `lsload`

   The output contains one line for each host in the cluster. The status should be ok for all hosts in your cluster.

Check the Remote Execution Server (RES)

You must use your user password using **lspasswd**.

1. Run a command on one LSF host, using the RES:

   `lsrun -v -m` *hostA hostname*
2. Run a command on a group of hosts, using the RES:

   `lsgrun -v -m "`*hostA hostB hostC*`"` *hostname*
3. Check for `OK` status on cross-cluster configuration information:

   `lsclusters -l`

LSF on EGO

LSF on EGO allows EGO to serve as the central resource broker, enabling enterprise applications to benefit from sharing of resources across the enterprise grid.

How to handle parameters in lsf.conf with corresponding parameters in ego.conf

When EGO is enabled, existing LSF parameters (parameter names beginning with LSB_ or LSF_) that are set only in `lsf.conf` operate as usual because LSF daemons and commands read both `lsf.conf` and `ego.conf`.

Some existing LSF parameters have corresponding EGO parameter names in `ego.conf` (`LSF_CONFDIR\lsf.conf` is a separate file from `LSF_CONFDIR\ego\`*cluster_name*`\kernel\ego.conf`). You can keep your existing LSF parameters in `lsf.conf`, or your can set the corresponding EGO parameters in `ego.conf` that have not already been set in `lsf.conf`.

You cannot set LSF parameters in `ego.conf`, but you can set the following EGO parameters related to LIM, PIM, and ELIM in either `lsf.conf` or `ego.conf`:

- EGO_DAEMONS_CPUS
- EGO_DEFINE_NCPUS
- EGO_SLAVE_CTRL_REMOTE_HOST
- EGO_WORKDIR

- EGO_PIM_SWAP_REPORT

You cannot set any other EGO parameters (parameter names beginning with EGO_) in `lsf.conf`. If EGO is not enabled, you can only set these parameters in `lsf.conf`.

Note:

If you specify a parameter in `lsf.conf` and you also specify the corresponding parameter in `ego.conf`, the parameter value in `ego.conf` takes precedence over the conflicting parameter in `lsf.conf`.

If the parameter is not set in either `lsf.conf` or `ego.conf`, the default takes effect depending on whether EGO is enabled. If EGO is not enabled, then the LSF default takes effect. If EGO is enabled, the EGO default takes effect. In most cases, the default is the same.

Some parameters in `lsf.conf` do not have exactly the same behavior, valid values, syntax, or default value as the corresponding parameter in `ego.conf`, so in general, you should not set them in both files. If you need LSF parameters for backwards compatibility, you should set them only in `lsf.conf`.

If you have LSF 6.2 hosts in your cluster, they can only read `lsf.conf`, so you must set LSF parameters only in `lsf.conf`.

LSF and EGO corresponding parameters

The following table summarizes existing LSF parameters that have corresponding EGO parameter names. You must continue to set other LSF parameters in `lsf.conf`.

lsf.conf parameter	ego.conf parameter
LSF_API_CONNTIMEOUT	EGO_LIM_CONNTIMEOUT
LSF_API_RECVTIMEOUT	EGO_LIM_RECVTIMEOUT
LSF_CLUSTER_ID (Windows)	EGO_CLUSTER_ID (Windows)
LSF_CONF_RETRY_INT	EGO_CONF_RETRY_INT
LSF_CONF_RETRY_MAX	EGO_CONF_RETRY_MAX
LSF_DEBUG_LIM	EGO_DEBUG_LIM
LSF_DHPC_ENV	EGO_DHPC_ENV
LSF_DYNAMIC_HOST_TIMEOUT	EGO_DYNAMIC_HOST_TIMEOUT
LSF_DYNAMIC_HOST_WAIT_TIME	EGO_DYNAMIC_HOST_WAIT_TIME
LSF_ENABLE_DUALCORE	EGO_ENABLE_DUALCORE
LSF_GET_CONF	EGO_GET_CONF
LSF_GETCONF_MAX	EGO_GETCONF_MAX
LSF_LIM_DEBUG	EGO_LIM_DEBUG
LSF_LIM_PORT	EGO_LIM_PORT
LSF_LOCAL_RESOURCES	EGO_LOCAL_RESOURCES
LSF_LOG_MASK	EGO_LOG_MASK

lsf.conf parameter	ego.conf parameter
LSF_MASTER_LIST	EGO_MASTER_LIST
LSF_PIM_INFODIR	EGO_PIM_INFODIR
LSF_PIM_SLEEPTIME	EGO_PIM_SLEEPTIME

Parameters that have changed in LSF

The default for LSF_LIM_PORT has changed to accommodate EGO default port configuration. On EGO, default ports start with lim at 7869, and are numbered consecutively for **pem**, **vemkd**, and **egosc**.

This is different from previous LSF releases where the default LSF_LIM_PORT was 6879. **res**, **sbatchd**, and **mbatchd** continue to use the default pre-version 7 ports 6878, 6881, and 6882.

Upgrade installation preserves existing port settings for **lim**, **res**, **sbatchd**, and **mbatchd**. EGO **pem**, **vemkd**, and **egosc** use default EGO ports starting at 7870, if they do not conflict with existing **lim**, **res**, **sbatchd**, and **mbatchd** ports.

EGO connection ports and base port

On every host, a set of connection ports must be free for use by LSF and EGO components.

LSF and EGO require exclusive use of certain ports for communication. EGO uses the same four consecutive ports on every host in the cluster. The first of these is called the base port.

The default EGO base connection port is 7869. By default, EGO uses four consecutive ports starting from the base port. By default, EGO uses ports 7869-7872.

The ports can be customized by customizing the base port. For example, if the base port is 6880, EGO uses ports 6880-6883.

LSF and EGO needs the same ports on every host, so you must specify the same base port on every host.

Check LSF

The LIM and **mbatchd** must be running on the master host and on the submission host (the host from which you run the command).

1. Verify the LSF daemon configuration:

   C:\LSF_9.1.3>`badmin ckconfig -v`

   The following message appears: `No errors found.`
2. Run some basic commands and check the status: OK (hosts) and Open:Active (queues):

   ```
   bhosts
   bqueues
   ```
3. Display the default queue:

   ```
   C:\lsf\bin>bparams
   ```
4. Submit a test job to the default queue named normal:

```
C:\lsf\9.1.3\bin> bsub sleep 60
Job <1> is submitted to default queue <normal>.
```

Note that the LSF installer for Windows sets "Log on as batch job" rights on Windows execution hosts as a basic requirement to run jobs.

5. Display the job status:

```
C:\lsf\9.1.3\bin> bjobs
```

If all hosts are busy, the job is not started immediately and the STAT column says PEND. The job `sleep 60` should take one minute to run. When the job completes, LSF sends mail reporting the job completion.

Chapter 2. IBM Platform LSF Default User Mapping

The default user mapping in LSF has no effect on a UNIX-only cluster. You do not need to understand this feature unless your cluster includes Windows hosts.

About LSF default user mapping

The default user mapping determines whether you can specify a Windows user in LSF by the user name alone. In a mixed cluster, it also specifies whether a Windows user account maps to a UNIX account of the same name, to allow cross-platform operation.

How LSF default user mapping works

If you specify an LSF user domain, the default user mapping is enabled. For a multiple-domain Windows environment on a UNIX-Windows mixed cluster, you can specify an unlimited number of Windows domains as the LSF user domain.

When the default user mapping is enabled,

- A user name specified without a domain is interpreted (on a Windows host) as belonging to the LSF user domain
- A user name specified with the domain name of the LSF user domain is stripped of the domain name

Mixed cluster

In a mixed UNIX-Windows environment, if your Windows account in the LSF user domain has the same user name as your UNIX account, LSF's default user mapping lets LSF schedule and track jobs from both accounts as if they belong to a single user. On the execution host, LSF automatically runs the job using whichever of the two accounts is appropriate for that host.

To submit cross-platform jobs when your accounts have different user names in different environments, you should configure user account mapping for individual users.

Multiple domain accounts

To run jobs, the existing domain trust relationships apply in LSF, so if the execution domain trusts the submission domain, your job can run in the execution domain under your submission account.

If a user domain is...	Then LSF treats the Windows and UNIX user as...
Specified by the parameter LSF_USER_DOMAIN	The same user
Not specified by the parameter LSF_USER_DOMAIN	Different users

Accounts with the same user name in different domains are still treated as separate users by LSF.

You can use the environment variable LSF_EXECUTE_DOMAIN to specify only one of the domains listed in LSF_USER_DOMAIN. When you specify an execution

© Copyright IBM Corp. 1992, 2014

domain, LSF runs the job using the specified domain user account, without trying all of the domain accounts in the order listed in LSF_USER_DOMAIN.

Local accounts

If your local account has the same user name and password on every Windows host, LSF's default user mapping lets LSF schedule and track jobs from all hosts as if they belong to a single user. On the execution host, LSF automatically runs the job using the local user account.

If your accounts have different user names in different environments, you should configure user account mapping.

Installation examples

In the following examples, assume you are User1, and you have a valid user account in 3 Windows domains as well as a valid UNIX account. Not all the accounts can be used with LSF. Depending on the type of cluster, and the way you install the cluster, here are the different ways that LSF is configured:

Install or upgrade a UNIX-only cluster

No mapping. You have one UNIX account, and LSF recognizes 1 user:

- `user1` (UNIX account)

Install a new Windows-only cluster

No mapping. You have 3 Windows accounts. For purposes of fairshare, per-user job slot limits, displaying statistical data, and so on, LSF recognizes 3 separate users:

- `DOMAINA\user1`
- `DOMAINB\user1`
- `DOMAINC\user1`

Create a new UNIX-Windows cluster

You must enable default user mapping for one of your Windows accounts (such as Domain A) so that you can run cross-platform jobs between UNIX and Windows. LSF recognizes 3 separate users:

- `user1` (your UNIX and Domain A accounts are treated as a single LSF user)
- `DOMAINB\user1`
- `DOMAINC\user1`

If you never run cross-platform jobs, you might choose to disable default user mapping by not specifying an LSF user domain. LSF then recognizes 4 separate users:

- `user1` (UNIX account)
- `DOMAINA\user1`
- `DOMAINB\user1`
- `DOMAINC\user1`

You can specify multiple domains when you define LSF_USER_DOMAIN, which will allow users to submit jobs from a UNIX host in a multiple-domain Windows environment.

Specify user names

In a Windows cluster or mixed UNIX-Windows cluster, in a domain environment, LSF users in different Windows domains might have the same user name. Because of this, LSF uses the Windows domain name with the user name, to differentiate the users.

User name only

When the default mapping is enabled, the user name alone specifies a user in the LSF user domain. The combination of a user name plus the domain name of the LSF user domain is not used in LSF.

Domain name with user name

Default mapping disabled

All Windows user accounts are specified using the domain name with the user name. There is no LSF user domain.

Default mapping enabled

User accounts in all domains except for the LSF user domain are specified using the domain name with the user name.

How to specify a user name with a domain name

Unless a Windows user account belongs to the LSF user domain (LSF_USER_DOMAIN in `lsf.conf`), the combination of domain name and user name specifies a Windows domain user in LSF. The syntax is:

`[`*`DOMAIN_NAME`*`|.]\`*`user_name`*

Type the domain name in capital letters. Use a period (.) instead of a domain name to specify a local account instead of a domain account.

UNIX systems interpret the single backslash as a special character, so on UNIX you have to use a double backslash to specify the domain name in the command line:

Windows

```
bjobs -u MYDOMAIN\user1
```

UNIX

```
bjobs -u MYDOMAIN\\user1
```

View user names

Use **bjobs -w** to view information about jobs and see the full name of a Windows user, including domain name.

When you run **bjobs**, the default is to truncate user names, and display the names of Windows users without the domain name.

Windows user authentication

LSF recognizes UNIX and Windows authentication environments, including different Windows domains and individual Windows workgroup hosts.

- In a Windows domain environment, user accounts are validated at the domain level, and your user account is valid on all hosts in your domain (and might be valid in other domains, if there is a trust relationship).
- In a Windows workgroup environment, each host authenticates the user account, so your local account is only valid on one host.

lspasswd command

You must use **lspasswd** or **wgpasswd** to register and update user names and passwords. The password must be 31 characters or less.

You can run **lspasswd** on Windows in a non-shared file system environment. You must define the parameter LSF_MASTER_LIST in `lsf.conf` so that jobs will run with the correct permissions. If this parameter is not defined, LSF assumes that the cluster uses a shared file system environment.

You can also run **lspasswd** to check that the password is valid for the specified user, or to remove a user entry from the password database.

Password problem notification on Windows

A Windows job may not be able to run because of a problem with the user's LSF password (entered and updated using **lspasswd**). If LSF does not recognize the password, the problem could be:

- The Windows user account password was never registered with LSF with **lspasswd**.
- The password in Windows changed but was not updated in LSF with **lspasswd**.

If a job is in PEND state and LSF cannot run it because of a password problem, by default, LSF puts the job into USUSP and then notifies the user via email. The user can fix the problem, and then use bresume to release the job from USUSP.

Configure LSF default user mapping

Whenever you make any change to default user mapping, you affect users in the old LSF user domain and in the new LSF user domain. If you specify a new LSF user domain, users in both domains will have to use **lspasswd** to register their new names and passwords.

If users in the old and new LSF user domain have the same user name (such as `olddomain\user1` and `newdomain\user1`), then the `user1` account is already registered with LSF, and the user from the new LSF user domain has to change the password. To change the password, he must input the current password, which was set by the old user.

To enable or modify default user mapping after you install LSF, set LSF_USER_DOMAIN in `lsf.conf` and specify the LSF user domain:

```
LSF_USER_DOMAIN=DomainA
```

- You can also specify multiple domains:

  ```
  LSF_USER_DOMAIN=DomainA:DomainB:DomainC
  ```

Depending on the cluster configuration, you might have to redefine the service accounts, cluster administrators, queue administrators, user group memberships, and so on, so that your cluster remains operational after you restart the cluster.

Syntax substitution for Windows user names

In *Administering IBM Platform LSF* and other LSF documentation, a user name is represented by the syntax:

user_name

If your cluster includes Windows hosts, the full syntax for a user account on Windows is:

[*DOMAIN_NAME*\ | .\]*user_name*

Always type the domain name in capital letters.

LSF commands

In the following LSF commands, use the full syntax to specify a user name.

- **bchkpnt**
- **bdel**
- **bhist**
- **bjobs**
- **bkill**
- **bmig**
- **bmod**
- **brequeue**
- **bresume**
- **bstop**
- **bsub**
- **bswitch**
- **busers**
- **lsacct**
- **lspasswd**

LSF files

In the following configuration files and parameters, use the full syntax to specify a user name.

- `lsb.hosts`
  - USER_SHARES
- `lsb.params`
  - SYSTEM_MAPPING_ACCOUNT
- `lsb.queues`
  - ADMINISTRATORS
  - FAIRSHARE
  - USERS
- `lsb.users`
  - GROUP_MEMBER
  - USER_SHARES
  - USER_NAME

  - LOCAL
  - REMOTE
- `lsf.cluster.`*`cluster_name`*
  - ADMINISTRATORS
- `lsf.conf`
  - LSF_SHELL_AT_USERS
- `lsf.sudoers`
  - LSF_EAUTH_USER
  - LSF_EEXEC_USER
  - LSF_STARTUP_USERS
  - LSB_PRE_POST_EXEC_USER

Chapter 3. Environment

Job execution environment

By default, LSF transfers environment variables from the submission to the execution host. However, some environment variables do not make sense when transferred.

How LSF sets the job execution environment

When submitting a job from a Windows to a UNIX machine, the **-L** option of **bsub** can be used to reinitialize the environment variables. If submitting a job from a UNIX machine to a Windows machine, you can set the environment variables explicitly in your job script.

PATH environment variable on UNIX and Windows

LSF automatically resets the PATH on the execution host if the submission host is of a different type. If the submission host is Windows and the execution host is UNIX, the PATH variable is set to `/bin:/usr/bin:/sbin:/usr/sbin` and the path to the LSF `bin` directory is appended to it.

If the submission host is UNIX and the execution host is Windows, the PATH variable is set to the system PATH variable with the path to the LSF `bin` directory appended to it. LSF looks for the presence of the WINDIR variable in the job's environment to determine whether the job was submitted from a Windows or UNIX host. If WINDIR is present, it is assumed that the submission host was Windows; otherwise, the submission host is assumed to be a UNIX machine.

Environment variable handling on Windows

The following Windows environment variables are overridden based on the values on the execution host:

- COMPSPEC
- COMPUTERNAME
- NTRESKIT
- OS2LIBPATH
- PROCESSOR_ARCHITECTURE
- PROCESSOR_LEVEL
- SYSTEMDRIVE
- SYSTEMROOT
- WINDIR

  If the WINDIR on the submission and execution host are different, then the system PATH variable on the execution host is used instead of that from the submission host.

  Avoid using drive names in environment variables (especially the PATH variable) for drives that are connected over the network. It is preferable to use the UNC form of the path. This is because drive maps are shared between all users logged on to a particular machine. For example, if an interactive user has drive `F:` mapped to `\\serverX\share`, then any batch job will also see drive `F:`

© Copyright IBM Corp. 1992, 2014

mapped to \\serverX\share. However, drive F: might have been mapped to a different share on the submission host of the job.

Job starters can be used to perform more site-specific handling of environment variables.

Control the execution environment using job starters

If running jobs on a Windows execution host, you cannot use the command **bsub -L**. Instead, LSF provides two job starters that apply the user environment of the execution host.

By default, the job starter executables are installed in LSF_BINDIR. If you prefer to store them elsewhere, make sure they are in a directory that is included in the default PATH on the execution host.

For example, on Windows, put the job starter under %WINDIR%.

The source code for the job starters is installed in LSF_TOP\9.1.3\examples.

Use either of the two following starter scripts to run jobs on a Windows execution host:

- **preservestarter**: Preserves the default user environment of the execution host; does not include any submission host settings
- **augmentstarter**: Augments the default user environment of the execution host by adding settings from the submission host that are not already defined on the execution host

Chapter 4. Charting Resources with Windows Performance Monitor

LSF integrates with Windows Performance Monitor, so you can chart LSF cluster, host, queue, and job performance information. Windows Performance Monitor can also be used to trigger external commands when specified thresholds are exceeded.

A service called LSF Monitor passes information from LSF to the Windows Performance Monitor. LSF Monitor must be installed separately.

LSF Monitor statistics

Once installed, LSF Monitor automatically sends information to the Windows Performance Monitor. Use the Windows Performance Monitor to chart LSF performance information.

The host, queue, and job objects support multiple instances.

The following LSF information is available:

- Cluster information
- Host information
- Queue information
- Job information
- External information

Cluster information

- Number of available servers
- Number of unavailable servers
- Number of servers where an LSF daemon (**sbatchd** or RES service) is down
- Number of unlicensed servers
- Number of pending jobs in the cluster
- Number of running jobs in the cluster
- Number of suspended jobs in the cluster
- Number of sick jobs (jobs submitted with no password, jobs with job dependency never satisfied, and jobs pending more than 3 days)
- Response time of LIM (as measured by the time to make an `ls_load` call)
- Response time of **mbatchd** (as measured by the time to make an `lsb_queueinfo` call)

Host information

- Load indices: r15s, r15m, mem, swap, pg, ut
- Number of running jobs
- Number of suspended jobs
- Number of reserved job slots
- External load Indices

© Copyright IBM Corp. 1992, 2014

Queue information

- Number of pending jobs
- Number of running jobs
- Number of suspended jobs
- Number of reserved job slots

Job information

- CPU time used by the job
- Memory used by the job (for jobs running on UNIX only)
- Swap space used by the job (for jobs running on UNIX only)

External information

- Values of one or two external load indices (configured by the LSF administrator)

Install LSF Monitor

You must have a cluster running LSF version 4.0 or higher. You must install LSF Monitor on any LSF server or client host running Windows. The cluster can include UNIX hosts. You must specify a cluster administrator account and password.

The LSF Monitor setup program is installed with LSF (LSF Monitor is not supported on 64bit machines). Use **lsfmon -install** to actually install the LSF Monitor service:

1. Log on to a Windows host as an LSF user in an existing LSF cluster.
2. In a command prompt, type:

   ```
   lsfmon -install
   ```

   LSF Monitor is installed.
3. On the Windows Control Panel, click **Services**.

   The Services window opens.
4. Right-click **LSF Monitor** and click **Properties**.
5. In the Log On As section, deselect **System Account**, select **This Account**, and specify an LSF cluster administrator account (such as `Administrator`).
6. Type in the password twice and click **OK**.
7. In the Services window, select **LSF Monitor** and click **Start** to start the service.

Configure LSF Monitor

Back up your registry before you make any changes.

You can configure sample intervals for host, queue and job information along with external load indices.

LSF Monitor periodically samples information from LSF and updates the Windows Performance Monitor.

By default, information is sampled at the following intervals:

- Host information = 30 seconds
- Queue information = 45 seconds
- Job information = 60 seconds

1. Change the sample intervals for LSF host, job, or queue information by modifying the Windows Registry settings.
   a. Select the Registry subkey:

      `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LSFMonitor`

   b. Edit the appropriate value, and specify the new sample interval in seconds:
      - SampleIntervalHost
      - SampleIntervalJob
      - SampleIntervalQueue
2. Configure LSF Monitor to monitor external load indices.
   a. Go to the Registry subkey `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LSFMonitor`.
   b. Specify the appropriate value and type the name of an external load index that is configured in your cluster:
      - ExternalLoadIndex1
      - ExternalLoadIndex2

Administer LSF Monitor

- Start or stop LSF Monitor.

  Use the Windows Control Panel to start or stop the LSF Monitor service.
- Use the Windows Event Viewer to view the Windows event log.

  Errors related to LSF API calls and the operation of LSF services are logged to the Windows event log.
- Uninstall LSF Monitor. From a command prompt, type:

  ```
  lsfmon -remove
  ```

  This command stops the LSF Monitor service if it is running, then removes it and removes related information from the Windows Registry.

Chapter 5. Dynamic IP Addressing for LSF Hosts

About dynamic host configuration protocol (DHCP)

DHCP (Dynamic Host Configuration Protocol) enables individual computers on an IP network to extract their configurations from particular machines (DHCP servers) that have no exact information about the individual computers until they request the information. This reduces the work necessary to administer a large IP network. The most significant piece of information distributed in this manner is the IP address.

How LSF works with dynamic IP addressing

LSF hosts running Windows can be configured as DHCP clients, which means their IP address is dynamic. Users who dial in or connect from a remote location might be assigned a different IP address with each connection.

The DHCP server issues an IP address to the LSF host, but LSF gets the IP address from DNS (Domain Name System). A WINS (Windows Internet Naming Service) server synchronizes information between the DHCP and DNS servers.

The IP address should not be changed while there are active TCP/IP connections with the host, for example, while installing LSF or running LSF commands. Normally, the IP address is maintained until the host is restarted or until the network connection is broken.

If an LSF client host is assigned a new IP address, wait for WINS to update DNS before using that host to run LSF.

LSF client hosts

LSF client hosts can be DHCP clients and can change their IP addresses anytime in a running cluster.

LSF server hosts

Installing dynamic hosts on Windows allows support for dynamic IP addressing for LSF server hosts using DHCP.

LSF server hosts can be DHCP clients and can change their IP addresses anytime in a running cluster. The master host also saves the slave host IP address.

Set up DHCP clients

To use DHCP with LSF, your system must include all of the following:

- DHCP server
- WINS server
- DNS server
- LSF hosts acting as DHCP clients

  Install Microsoft DNS server and WINS server on the same machine.

1. Configure a short cache timeout value on the WINS server.

© Copyright IBM Corp. 1992, 2014

The Cache Timeout Value for the WINS Lookup of the DNS should be as short as possible (the 10-minute default may be acceptable, but this should not be increased).

2. Enable dynamic IP addressing for the LSF cluster.
   a. Configure the following parameter in `lsf.conf`: LSF_DHCP_ENV=Y
   b. Reconfigure the cluster:

      ```
      lsadmin reconfig
      badmin reconfig
      ```

      LSF checks for any configuration errors. If no fatal errors are found, you are asked to confirm reconfiguration. If fatal errors are found, reconfiguration is aborted.

Chapter 6. Displaying a Graphical User Interface in LSF with Microsoft Terminal Services

How LSF works with Remote Desktop Services

Use Microsoft Terminal Services to run jobs to display a GUI (graphic user interface) on remote hosts in LSF.

Environment variables

The following environment variables are available for Remote Desktop Services jobs:

LSF_LOGON_DESKTOP

When LSF_LOGON_DESKTOP=1, jobs run in interactive foreground sessions. This allows GUIs to be displayed on the submission host. If this parameter is not defined, jobs run in the background.

LSB_TSJOB

When the LSB_TSJOB variable is defined to any value, it indicates to LSF that the job is a Terminal Services job.

LSF_TS_LOGON_TIME

Specifies the time to create a Windows Remote Desktop Service session. You should configure LSF_TS_LOGON_TIME according to your network environment. This parameter only affects Terminal Services jobs dispatched to the Windows server version host, not Terminal Services jobs on the Windows client version host.

The default, 300000 milliseconds is suitable for most environments. For a congested network, set LSF_TS_LOGON_TIME=1000000

Configuration files

To dispatch a Terminal Services jobs to Windows client hosts, set the following parameters in lsf.conf after installation, then restart the LIM (by running **lsadmin limrestart all**) and restart the **TSJobHelper** Windows service on the execution hosts and helper hosts to apply the changes to these parameters:

LSB_TSJOBS_HELPER_HOSTS

Lists the TS job helper hosts. Specify a list of host names separated by blank space. Helper hosts must be LSF servers in the LSF cluster. Configure a maximum of 256 hosts in the list.

The local helper service will select one host from the list and send requests to the helper to create a user session. If the host fails to create the user session, the next helper host in the list will be tried. The local helper service will not select itself as helper host.

For stability, you should configure one helper host for every 40 execution hosts.

TSJobHelper must be running under the LocalSystem account. The TS job helper service has two functions:

© Copyright IBM Corp. 1992, 2014

- Help create a user session. The local helper service running on the job execution host sends requests to the remote helper service running on a helper host. The remote helper host will call back to trigger creation of a user session.
- Help launch the LSF job across sessions. On the job execution host, the service accepts job startup information from child **sbatchd** and launches the job RES across sessions. The job will run in the job execution session for the user.

To install the TSJobHelper service, run "TSJobHelper -i" from the command line. To uninstall the service, run "TSJobHelper -u". TSJobHelper is located under LSF_TOP\9.1.3\etc.

LSB_TSJOBS_HELPER_PORT

This is a service port to use for communication with TSJobHelper. TSJobHelper uses this port to communicate with the helper hosts.

LSB_TSJOBS_HELPER_TIMEOUT

The maximum time out that the local TSJobHelper service waits for a helper host to reply. After time out, the local service tries the next helper host in the LSB_TSJOBS_HELPER_HOSTS list.

Configure post-execution job processing

To dispatch a Terminal Services jobs to Windows client hosts, enable post-execution job processing as part of the job. You may enable post-execution job processing to dispatch Terminal Services jobs to Windows client hosts for the following:

- Cluster: To enable the dispatching of Terminal Services jobs in the cluster to Windows client hosts, set `JOB_INCLUDE_POSTPROC=y` in the `lsb.params` file, then reconfigure mbatchd (by running **badmin reconfig**) to apply the change to this parameter.
- Application: To enable the dispatching of Terminal Services jobs in a specific application to Windows client hosts, set `JOB_INCLUDE_POSTPROC=y` for the specific application profile in the `lsb.applications` file, then reconfigure mbatchd (by running **badmin reconfig**) to apply the change to this parameter.
- Job: To enable the dispatching of a specific Terminal Services jobs to Windows client hosts, set the `JOB_INCLUDE_POSTPROC` environment variable, then submit the Terminal Services job.

Job submission

1. Submit the job with **tssub** instead of **bsub**. **tssub** is a wrapper around the **bsub** command which only submits jobs to hosts that have the msts resource.
2. **tssub** sets the LSB_TSJOB and LSF_LOGON_DESKTOP environment variables. These variables are then transferred to the execution host.
   - If the job is dispatched to a host in which Terminal Services is not installed or properly configured, the job is set to the PEND state and a pending reason is written in `sbatchd.log.`*`host_name`*.
   - If **tssub -I** is specified, a terminal display is visible on the submission host after the job has been started.
   - If the job is not a GUI job, LSF runs a command window and output is displayed in the command window when something is written to `stdout`.
   - Pre- and post-execution commands are executed within the session in which **sbatchd** runs. The job does not complete until post-execution commands complete.

3. View job output with the command **tspeek**. If the terminal window is closed, the job remains running. You can reconnect to view the job with **tspeek**.

Limitations

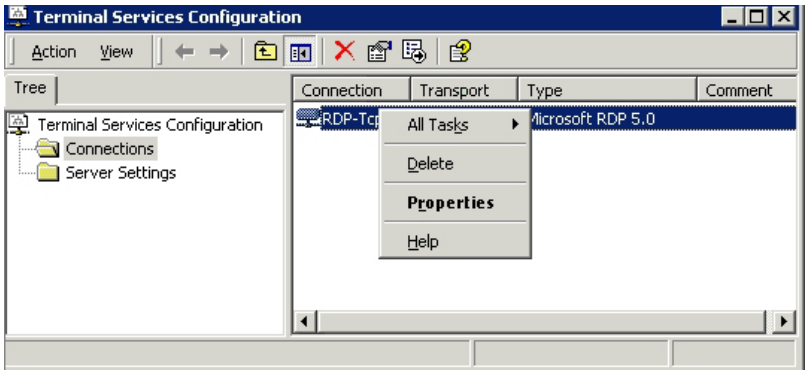
- A job submitted as a Remote Desktop Services job cannot be modified to become a non-Remote Desktop Services job with **bmod**.
- The bsub option **-o** *out_file* is not supported for **tssub**.
- Compound resource requirements are not supported for Remote Desktop Services jobs.
- Only Windows **bsub** options are supported for **tssub**. For example, you cannot use the options **-Ip**, **-Is**, **-L** *login_shell* of **bsub** with **tssub**.
- If user mapping is defined, the user who invokes **tspeek** must have the required privileges to access the session.
- If you are planning to run TSJobHelper service on a Windows host, LSF must apply the non-shared configuration directory when installing.
- Platform MultiCluster is not supported.

Requirements

- All submission hosts must have the Remote Desktop Services Full Client Windows Installer (MSI) package installed and enabled. This package contains the required Microsoft Terminal Services Advanced Client ActiveX Control. Download it from the Microsoft web site.
- To run TS jobs on Windows hosts, you must install Terminal Server Role.
- All execution server hosts must have the Remote Desktop Access enabled.
- Remote Desktop Services server hosts can be in a different domain from submission hosts as long as the user can be authenticated by both domains.
- The service account on Remote Desktop Services server hosts must be a member of the local administrators group.
- Your LSF cluster must be working properly.
- All submission (client) hosts need the ActiveX control.

Configure Remote Desktop Services for LSF

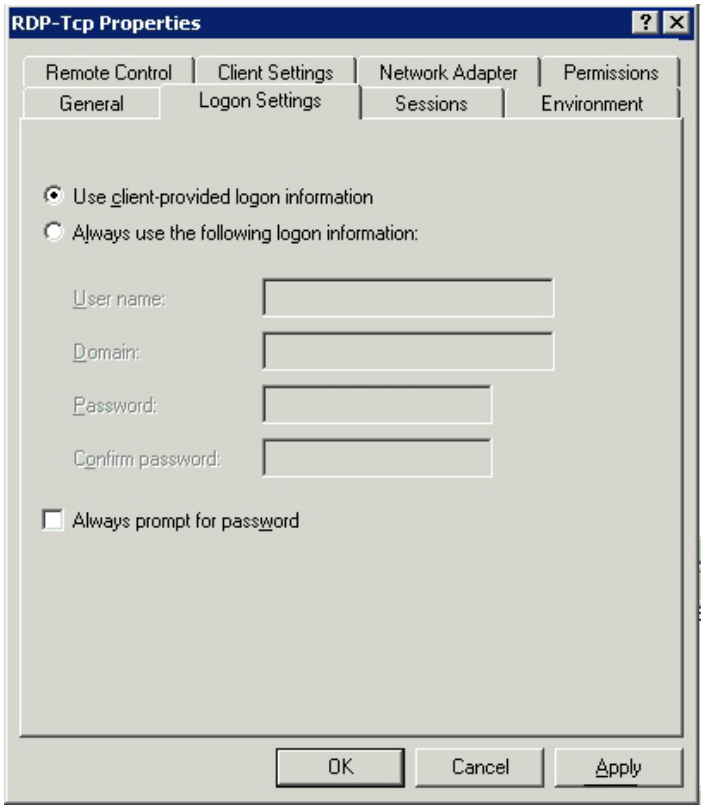
1. Start the Microsoft Management Console (MMC) Terminal Services Configuration snap-in (**Start** > **Programs** > **Administrative Tools** > **Terminal Services Configuration**).



2. Right-click the configuration for which you want to disable the default password setting, and select **Properties**.

3. Change the logon information.
   a. Select **Logon Settings** > **Use client provided logon information**.
      This ensures you are not using a predefined user to log on to Remote Desktop Services.
   b. Clear the Always prompt for password check box.

   Future connections no longer force a password entry.



4. Click **Apply**, then click **OK** and close the dialog.

Configure Windows Server 2008 for TS jobs

To launch Terminal Services jobs on a Windows 2008 server host, you need to configure `lstsmgr.exe` (an LSF binary) and `tscon.exe` (a Windows executable file) as allowed by the RemoteApp programs in RemoteApp Manager on the Windows 2008 server host.

To install the Terminal Server role service:

1. Open Server Manager (**Start** > **Programs** > **Administrative Tools** > **Server Manager**).
2. In the left pane, right-click **Roles**, and then click **Add Roles**.
3. In the **Add Roles** wizard, on the Before You Begin page, click **Next**.
4. In the Select Server Roles page, under **Roles**, select the **Terminal Services** check box. Click **Next**.
5. In the **Terminal Services** page, click **Next**.
6. In the Select Role Services page, select the **Terminal Services** check box, and then click **Next**.
7. In the Uninstall and Reinstall Applications for Compatibility page, click **Next**.

8. In the Specify Authentication Method for Terminal Server page, select the appropriate authentication method for the terminal server, and then click **Next**.
9. In the Specify Licensing Mode page, select the appropriate licensing mode for the terminal server, and then click **Next**.
10. In the Select User Groups Allowed Access To This Terminal Server page, add the users or user groups to which you want to remotely connect to this terminal server, and then click **Next**.
11. In the Confirm Installation Selections page, verify that the Terminal Server role service will be installed, and then click **Install**. On the Installation Progress page, installation progress will be noted.

Optionally, you can check the RemoteApp configuration to see if `lstsmgr.exe` and `tscon.exe` are listed in the RemoteApp allowed program list after LSF is installed:

1. Open TS Remote Manager (**Start** > **Programs** > **Administrative Tools** > **Terminal Services**, and then click TS RemoteApp Manager).
2. In the **Actions** side bar click **Add RemoteApp Programs**, then click **Next**.
3. In the **RemoteApp** wizard, click **Browse**, then select `lstsmgr.exe` (in `<LSF_TOP>\etc` in the list).
4. Select `C:\Windows\system32\tscon.exe` from the list. Click **Next** and **Finish**.
5. Right click the items in the RemoteApp Programs. Click **Properties** and select **Allow any command-line arguments**.

Alternatively, you can change Terminal Services settings and select **Allow users to start both listed and unlisted programs on initial connection**.

Configure Windows Server 2012 for TS jobs

Windows Server 2012 has security enhancements over Windows Server 2008, so to enable Terminal Service jobs for Windows Server 2012:

1. From the Control Panel, select **System and Security** > **System** > **Remote settings**.
2. Select **Allow remote connections to this computer**.
3. Make sure **Allow connections only from computer running Remote Desktop with Network Level Authentication** is not checked.
4. Ensure the following registry settings:
   - `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Terminal Server\TsAppAllowList\Applications\lstsmgr`
   - `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Terminal Server\TsAppAllowList\Applications\tscon`
5. Set `CommandLineSetting` to 1

Configure LSF to run Remote Desktop Services jobs

1. Define the msts static resource.

   Edit `LSF_CONFDIR\lsf.shared` and define the msts static resource. Note that the resource name must be msts and values for the resource must be exactly as shown.

```
Begin Resource
RESOURCENAME TYPE  INTERVAL INCREASING DESCRIPTION
...
msts         Boolean  ()    ()        (Windows Terminal Server)
...
End Resource
```

2. Add the msts resource to hosts.

   Edit LSF_CONFDIR\lsf.cluster.*cluster_name* and add the msts resource to each host on which Terminal Server is installed. For example:

```
Begin Host
HOSTNAME   model   type   server r1m pg tmp RESOURCES
...
hostA      !       NTX86    1     -   -   - (msts)
...
End Host
```

3. (Optional) Create job starters to preserve a user's environment settings.

   You may need to create a job starter script to preserve a user's environment settings on the execution host.

Use Windows Remote Desktop Services Job Support on Windows client version host

This feature only supports one Remote Desktop Services job running on each Windows host. Additional jobs will be in PEND state, causing mbschd to schedule the extra Remote Desktop Services jobs to the host back and forth. To avoid this, set a static numeric resource on each Windows host. For example, configure a static numeric resource, like tsjob and set it to 1 for each Windows host. The msts resource must also be configured.

To configure a static resource:

1. Add the following line in the lsf.shared file "Resource" section:

   ```
   tsjob Numeric () N (tsjob counter)
   ```

2. Add the following line in the lsf.cluster file in the "ResourceMap" section:

   ```
   tsjob (1@[win])
   ```

   In this example, win is the Windows host name.

A Remote Desktop job will fail on Windows when there is any user logged on the computer before the job is submitted. To avoid this, you should add ls==0 to the job resource requirement.

To submit a job, use:

```
tssub -R "ls==0 rusage[tsjob=1]" app
```

When the job runs, use tspeek to check job display from a Windows host or a Linux host with rdesktop installed.

Submit LSF jobs to Terminal Services hosts (tssub)

1. Submit a job to a host with Remote Desktop Services installed by using the **tssub** command:

   ```
   tssub myjob
   ```

   The bsub option **-o** *out_file* is not supported for **tssub**.

Only Windows **bsub** options are supported for **tssub**. For example, you cannot use the options **-Ip**, **-Is**, **-L** *login_shell* of **bsub** with **tssub**.

2. View job output:

   tspeek *job_ID*

   You can use **tspeek** from any LSF Windows to view the output of a Remote Desktop Services job:

   You can also use **tspeek** to monitor job output from a Linux host with **rdesktop** installed. You cannot use **tspeek** to monitor job output from UNIX.

3. Monitor the job:

   ```
   bjobs -l
   ```

   If you use to monitor the job, you see a message:

   ```
   External Message 2 was posted from LSF\lsfadmin to message box 2
   ```

   The body of the message contains the ID of the terminal session that was created.

Limit the number of Remote Desktop Services jobs on a host

The msts resource indicates to LSF whether an execution host has Remote Desktop Server installed or not.

To limit the number of Remote Desktop Services jobs that run on a host and keep track of how many jobs are running, define a numeric resource in addition to the msts boolean resource. Alternatively, use an elim to report how many terminal servers are available for each host. Microsoft Windows laptop class Operating Systems (for example, Windows XP, Windows Vista, Windows 7, etc.) only support a single Remote Desktop session, except Windows XP Home Edition. Windows XP Home edition does not support Remote Desktop Services and can not be used to submit or run Graphical User jobs within LSF.

1. Configure a numeric Remote Desktop Server resource.

   Define the resource in LSF_CONFDIR\lsf.shared. For example:

   ```
   Begin Resource
   RESOURCENAME TYPE INTERVAL INCREASING DESCRIPTION
   ...
   term_server  Numeric 60   N      (Terminal Server)
   ...
   End Resource
   ```

2. Submit a job with **rusage** . When submitting a job, use the rusage resource requirement string:

   ```
   tssub -R"rusage[term_server=1]" myjob
   ```

Submit a Remote Desktop Services job from UNIX

In mixed cluster environments, it is possible to submit a Remote Desktop Services job with **bsub** from a UNIX host. You can use **tspeek** to monitor job output from a Linux host with **rdesktop** installed. You cannot use **tspeek** to monitor job output from UNIX.

1. On the UNIX submission host, define the environment variables LSF_LOGON_DESKTOP=1 and LSB_TSJOB=1.
   - When LSF_LOGON_DESKTOP=1, it allows GUIs to be displayed on the submission host.
   - When the LSB_TSJOB variable is defined to any value, it indicates the job is a Remote Desktop Services job.

2. Submit the job with **bsub** and indicate the **msts** resource requirement. For example:

```
bsub -R"msts" myjob
```

Chapter 7. Installing LSF in a Mixed Cluster

Set up a Linux cluster with Windows compute nodes

Complete the following steps to set up a Linux cluster with Windows compute nodes.

1. Install the Linux cluster on a shared file system.
2. Configure the Linux cluster.
3. Restart the LSF cluster.
4. Install the Windows compute node.
5. Start the Windows compute node.

Install the Linux cluster on a shared file system

Install the Linux cluster as described in *Installing IBM Platform LSF on UNIX and Linux*, with exceptions to allow for Windows compute nodes.
Edit `install.conf` and specify the following:

1. Enable dynamic hosts.

   Enable or add the following line:

   `ENABLE_DYNAMIC_HOSTS=Y`
2. Optional. Allow EGO to control the LSF daemons.

   Enable or add the following line:

   `EGO_DAEMON_CONTROL=Y`
3. Specify the cluster administrator.

   `LSF_ADMIN=`*`user_account`*

   For example:

   `LSF_ADMIN=lsfadmin`
4. Specify the installation directory.

   `LSF_TOP=`*`directory`*

   For example:

   `LSF_TOP=$SHARE/LSF_9.1.3`

Configure the Linux cluster

1. Optional. If you allowed EGO to control the LSF daemons, add Windows compute node information to the LSF service configuration files.
   a. Edit `LSF_TOP/conf/ego/`*`cluster_name`*`/eservice/esc/conf/services/res.xml`
   b. Navigate to the section with the correct Windows host type.

      For 64-bit compute hosts, navigate to NTX64. For IA hosts, navigate to NTIA64. For other Windows compute hosts, navigate to NTX86.
   c. Add the proper Windows compute node information for the **`Command`** and **`ExecutionUser`** tags.

      For example:

```
<sc:ActivityDescription>
  <ego:Attribute name="hostType" type="xsd:string">NTX86</ego:Attribute>
  <ego:ActivitySpecification>
```

© Copyright IBM Corp. 1992, 2014

```
  <ego:Command>C:\LSF_9.1.3\9.1.3\etc\res.exe -3</ego:Command>
  <ego:ExecutionUser>LSF\lsfadmin</ego:ExecutionUser>
  <ego:EnvironmentVariable name="LSF_ENVDIR">C:\LSF_9.1.3\conf</ego:EnvironmentVariable>
  ...
```

d. Edit LSF_TOP/conf/ego/*cluster_name*/eservice/esc/conf/services/sbatchd.xml

e. Navigate to the section with the correct Windows host type.

For 64-bit compute hosts, navigate to **NTX64**. For IA hosts, navigate to **NTIA64**. For other Windows compute hosts, navigate to **NTX86**.

f. Add the proper Windows compute node information for the **Command** and **ExecutionUser** tags.

For example:

```
<sc:ActivityDescription>
  <ego:Attribute name="hostType" type="xsd:string">NTX86</ego:Attribute>
  <ego:ActivitySpecification>
  <ego:Command>C:\LSF_9.1.3\9.1.3\etc\sbatchd.exe -3</ego:Command>
  <ego:ExecutionUser>LSF\lsfadmin</ego:ExecutionUser>
  ...
```

2. Add the Windows cluster administrator account to the your cluster file.
   a. Edit LSF_CONFDIR/lsf.cluster.*cluster_name*
   b. In the ClusterAdmins section, add LSF\lsfadmin to the Administrators list.

   For example:

   ```
   Begin ClusterAdmins
   Administrators = lsfadmin LSF\lsfadmin
   End ClusterAdmins
   ```

3. Add the LSF user domain to the lsf.conf file.
   a. Edit LSF_TOP/conf/lsf.conf
   b. Add the LSF user domain.

   LSF_USER_DOMAIN=*lsf_user_domain*

   For example:

   LSF_USER_DOMAIN=LSF

4. Register the Windows execution password to your Linux cluster.
   a. If not using EGO to control the LSF daemons:
      - Register Windows user passwords to your cluster password file for all users submitting jobs to LSF.

      lspasswd -u *"domain\admin"* -p *password*

      For example:

      lspasswd -u "LSF\lsfadmin" -p lsfpasswd

      Passwords must be 31 characters or less.

   b. If using EGO to control the LSF daemons:
      - Log on to any host in the cluster as egoadmin.
      - Log on to EGO as the cluster administrator.

      For example:

      egosh user logon -u Admin -x mypasswd

      - Register Windows user passwords to your cluster password file for all users submitting jobs to LSF.

      egosh ego execpasswd -u "*domain\admin*" -x *password* -noverify

      For example:

      egosh ego execpasswd -u "LSF\lsfadmin" -x lsfpasswd -noverify

The password must be 31 characters or less.

The `-noverify` option is required since only a Windows host can verify the password for a Windows user.

Restart the LSF cluster

Restart the LSF cluster.

```
lsfstartup
```

Install the Windows compute node

1. Install the Windows compute node as described in *Installing IBM Platform LSF on Windows*, with exceptions to be part of a Linux cluster.

   Specify the following options during installation:

   a. Specify the Linux master host as your master host name.

      `Master_Name=`*`linux_master_name`*

   b. If you allowed EGO to control the LSF daemons in your cluster master host, allow EGO to control the LSF daemons in your compute node.

      `EGO_DAEMON_CONTROL=Y`

   c. Specify the same port number as that of the Linux master host.

      `Port_Number=`*`base_port_number`*

   d. Specify the cluster administrator to be the same as on your master host.

      `LSF_ADMIN=`*`domain\user_account`*

      For example:

      `LSF_ADMIN=LSF\lsfadmin`

   e. Specify the installation directory.

      `LSF_TOP=`*`directory`*

      For example:

      `LSF_TOP=C:\LSF_9.1.3`

2. Register the Windows execution user password to your cluster password file.

   `lspasswd -u "`*`domain\admin`*`" -p `*`password`*

   For example:

   `lspasswd -u "LSF\lsfadmin" -p lsfpasswd`

   The password must be 31 characters or less.

Start the Windows compute node

1. Start the Windows compute node:

   ```
   lsadmin limstartup
   ```

2. If you did not allow EGO to control the LSF daemons, manually start the LSF services:

   ```
   lsfadmin resstartup
   badmin hstartup
   ```

Set up a Windows cluster with Linux compute nodes

Complete the following steps to set up a Windows cluster with Linux compute nodes if Ego is not controlling the LSF daemons.

1. Install the Windows cluster master host as described in the Windows installation guide.

a. Add the LSF user domain to the `lsf.conf` file.
   - Edit `C:\LSF9.1.3\conf\lsf.conf`
   - Add the LSF user domain.

     `LSF_USER_DOMAIN=`*`lsf_user_domain`*

     For example:

     `LSF_USER_DOMAIN=LSF`

b. Restart the Windows cluster.

   `lsfrestart`

c. Register Windows execution user passwords to your cluster password file for all users submitting jobs to LSF.

   `lspasswd -u "`*`domain\admin`*`" -p `*`password`*

   For example:

   `lspasswd -u "LSF\lsfadmin" -p lsfpasswd`

   Passwords must be 31 characters or less.

2. Install the Linux compute nodes as described in the Linux installation guide.
   a. Specify the Windows master host as your master host name.

      `Master_Name=`*`windows_master_name`*

   b. Start the Linux compute nodes

Set up a Windows cluster with Linux compute nodes and EGO controlling LSF daemons

Complete the following steps to set up a Windows cluster with Linux compute nodes and EGO controlling the LSF daemons.

1. Install the Windows cluster master host as described in the Windows installation guide and allow EGO to control the LSF daemons.
   a. Specify the following option:

      `EGO_DAEMON_CONTROL=Y`

   b. Add the LSF user domain to the `lsf.conf` file.
      - Edit `C:\LSF9.1.3\conf\lsf.conf`
      - Add the LSF user domain.

        `LSF_USER_DOMAIN=`*`lsf_user_domain`*

        For example:

        `LSF_USER_DOMAIN=LSF`

2. Install the Linux compute node as described in the Linux installation guide and allow EGO to control the LSF daemons.

   Edit `install.conf` and add the following line:

   `EGO_DAEMON_CONTROL=Y`

3. Restart the Windows cluster.

   `egosh ego restart`

Notices

This information was developed for products and services offered in the U.S.A.

IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web

© Copyright IBM Corp. 1992, 2014

sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Intellectual Property Law
Mail Station P300
2455 South Road,
Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application

programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

LSF®, Platform, and Platform Computing are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Privacy policy considerations

IBM Software products, including software as a service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software

Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, See IBM's Privacy Policy at http://www.ibm.com/privacy and IBM's Online Privacy Statement at http://www.ibm.com/privacy/details the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at http://www.ibm.com/software/info/product-privacy.

Printed in USA

SC27-5311-03