Linux commands for BESAdmin tool

Found on page 217 and forward (from BigFix Installation guide as of 092718)  
ftp://public.dhe.ibm.com/software/tivoli/IEM/9.5/Platform/BigFix\_Installation\_Guide.pdf

./BESAdmin.sh -syncmastheadandlicense

**Running the BigFix Administration Tool**

The installation script install.sh automatically downloads the IBM BigFix

Administration Tool bash shell script, BESAdmin.sh, in the /opt/BESServer/bin

directory. With this tool you can edit the masthead file, check the signatures of the

objects in the database, enable and disable enhanced security, resign all of the users

content in the database, rotate the server private key, configure the Console and

Web Reports login, resign the database content, and synchronize the masthead

with the updated license.

Run this script, as super user from the command prompt, using the following

syntax:

./BESAdmin.sh -*service* {*arguments*}

where *service* can be one of the following:

audittrailcleaner

changeprivatekeypassword

editmasthead

findinvalidsignatures

importlicense

minimumSupportedClient

minimumSupportedRelay

propagateAllOperatorSites

propertyidmapper

removecomputers

Chapter 10. Installing on Linux systems **209**

repair

reportencryption

resetDatabaseEpoch

resignsecuritydata

revokewebuicredentials

rotateserversigningkey

securitysettings

setadvancedoptions

setproxy

syncmastheadandlicense

updatepassword

**Note:** The notation <path+license.pvk> used in the command syntax stands for

*path\_to\_license\_file*/license.pvk.

Each service has the following *arguments*:

**audittrailcleaner**

You can run this service to remove historical data from the bfenterprise

database that is stored to serve as an audit trail. This audit trail slowly

increases in size over the lifetime of a BigFix deployment. The audit trail

contains deleted and earlier versions of Fixlets, tasks, baselines, properties,

mailbox files, actions, and analyses. The audit trail is not used by BigFix in any

way and can be deleted to reduce the database size. BigFix recommends that

you create a historic archive of the current database and save it to a secure

location before running this tool to preserve the audit trail, thus removing it

from the product database, but not completely deleting the history.

The service can count and delete the following sets of data:

v **Older Versions of Custom Authored Content** (-oldcontent): Every edit to

Fixlets, Tasks, Baselines, and Analyses, creates a new version, the earlier

versions can be deleted.

v **Older Versions of Actions** (-oldactions): Any time you stop or start an

Action, a new version is created; the earlier versions can be deleted.

v **Older Versions of relay.dat** (-oldrelaydatfile): Any time you install or

uninstall a new relay, a new version is created; the earlier versions can be

deleted.

v **Deleted Custom Authored Content (all versions)** (-deletedcontent): When

you delete a Fixlet, Task, Baseline, and Analysis using the console, the data

is marked as deleted in the database and preserved. The deleted content,

including all of the earlier versions, and the corresponding client reports can

be deleted.

v **Deleted Actions(all versions)** (-deletedactions): When you delete an action

using the console, the data is marked as deleted in the database and

preserved. The deleted actions, including all of the earlier versions, and the

corresponding client reports can be deleted.

v **Useless Action Results** (-uselessactionresults): Earlier versions of IBM

Endpoint Manager (before BigFix 7.2.4.60) might cause clients to report

ActionResults that were not used in any way but would use up space in the

database. These useless ActionResults can be deleted.

v **Orphaned sub-actions** (-orphanedsubactions): From multiple action groups

that were deleted.

v **Hidden Manual Computer Group Actions** (-hiddenactions): Manual

Computer Groups create hidden actions that add and remove computers to

**210** IBM BigFix: Installation Guide

and from groups and the actions can build up over time. This option deletes

actions after an expiration period (default 180 days) from when they were

created.

v **Older Version of Mailbox Files** (-deletedmailbox): Deleted Mailbox Files

are stored in a table in the database and can be removed.

v **Synchronizing BES Consoles** (-syncconsoles): The BigFix Console

maintains a local cache of the database that becomes not synchronized when

data is removed with this tool. To prevent this situation from happening, the

tool sets a flag in the database to force all BigFix Consoles to reload the

cache when the Console is started up.

v **Removing data older than** (-olderthan): Removes data earlier than a

specified date. The default value is 99 days.

v **Batched deletion** (-batchsize): Deleting large sets of data causes the SQL

transaction log to quickly increase in size, the log becomes temporarily

larger than the data being removed until the database is shrunk. Batched

deletion removes results in sets.

The syntax of this service changes depending on the action you specify:

./BESAdmin.sh -audittrailcleaner { -displaysettings | -run [delete\_data\_options] |

-schedule [delete\_data\_options] [scheduling options] | -preview [delete\_data\_options]

[preview options] }

./BESAdmin.sh -audittrailcleaner **-displaysettings**

./BESAdmin.sh -audittrailcleaner **-run** [ -oldcontent ] [ -oldactions ]

[ -oldrelaydatfile ] [ -deletedcontent ] [ -deletedactions ]

[ -uselessactionresults ] [ -orphanedsubactions ] [ -hiddenactions=<days> ]

[ -deletedmailbox ] [ -syncconsoles ] [ -olderthan=<days> ] [ -batchsize=<size> ]

./BESAdmin.sh -audittrailcleaner -sitePvkLocation=<path+license.pvk>

[ -sitePvkPassword=<password> ] **-schedule** [ [ -oldcontent ] [ -oldactions ]

[ -oldrelaydatfile ] [ -deletedcontent ] [ -deletedactions ] [ -uselessactionresults ]

[ -orphanedsubactions ] [ -hiddenactions=<days> ] [ -deletedmailbox ] [ -syncconsoles ]

[ -olderthan=<days> ] [ -batchsize=<size> ] [ -cleanstarttime=<yyyymmdd:hhmm>

[ -cleanperiodicinterval=<hours> ] ] | -disable ]

./BESAdmin.sh -audittrailcleaner **-preview** [ [ -oldcontent ] [ -oldactions ] [

-oldrelaydatfile ] [ -deletedcontent ] [ -deletedactions ] [ -uselessactionresults ] [

-orphanedsubactions ] [ -hiddenactions=<days> ] [ -deletedmailbox ] [ -olderthan=<days> | [ -scheduled ] ]

where:

v **displaysettings** shows the settings that are previously defined with the

schedule action.

v **run** runs the tool with the specified settings. Before you use this option,

check the settings that affect the database by using the **preview** action.

v **schedule** schedules the tool to run at the specified time at each specified

interval. To disable the schedule action, use the -disable option.

v **preview** shows the number of database rows that are affected by the

specified settings. If no setting is passed to the preview option, the preview

performs the count by setting all options to true and using the default

values for dates. Use the -scheduled option to preview the scheduled

settings.

For information about the cleanup tasks log files, see “Logging Cleanup Tasks

Activities” on page 145.

**changeprivatekeypassword**

You can use this service to be prompted for a new password to associate to the

license.pvk file. Use the following syntax to run the command:

Chapter 10. Installing on Linux systems **211**

./BESAdmin.sh -changeprivatekeypassword -sitePvkLocation=<path+license.pvk>

[ -sitePvkPassword=<password> ]

**editmasthead**

You can edit the masthead file by specifying the following parameters:

advGatherSchedule (optional, integer)

values:

0=Fifteen Minutes,

1=Half Hour, 2=Hour,

3=Eight Hours,

4=Half day,

5=Day,

6=Two Days,

7=Week,

8=Two Weeks,

9=Month,

10=Two Months

advController (optional, integer)

values:

0=console,

1=client,

2=nobody

advInitialLockState (optional, integer)

values:

0=Locked,

1=timed (specify duration),

2=Unlocked

advInitialLockDuration (optional, integer)

values:

( duration in seconds )

advActionLockExemptionURL (optional, string)

advRequireFIPScompliantCrypto (optional, boolean)

The syntax to run this service is:

./BESAdmin.sh -editmasthead -sitePvkLocation=<path+license.pvk>

[ -sitePvkPassword=<password> ][ -display ]

[ -advGatherSchedule=<0-10> ] [ -advController=<0-2> ]

[ -advInitialLockState=<0|2> | -advInitialLockState=1

-advInitialLockDuration=<num> ] [ -advActionLockExemptionURL=<url> ]

[ -advRequireFIPScompliantCrypto=<true|false> ]

For additional information, see Editing the Masthead on Linux systems in the

*IBM BigFix Configuration Guide*.

**findinvalidsignatures**

You can check the signatures of the objects in the database by specifying the

following parameters:

**-list (optional)**

Lists all invalid signatures that BESAdmin finds.

**-resignInvalidSignatures (optional)**

Attempts to resign any invalid signatures that BESAdmin finds.

**-deleteInvalidlySignedContent (optional)**

Deletes contents with invalid signatures.

For additional information about invalid signatures see http://www-

01.ibm.com/support/docview.wss?uid=swg21587965. The syntax to run this

service is:

./BESAdmin.sh -findinvalidsignatures

[ -list | -resignInvalidSignatures | -deleteInvalidlySignedContent ]

**212** IBM BigFix: Installation Guide

**importlicense**

You can use this service to import an updated license. This service allows you

to update the license manually in isolated IBM Endpoint Manager

environments.

./BESAdmin.sh -importlicense -sitePvkLocation=<path+license.pvk>

[ -sitePvkPassword=<password> ] -licenselocation=<path+license.crt>

The license.crt file contains the updated license to import.

**minimumSupportedClient**

This service defines the minimum version of the BigFix Agents that are used in

your BigFix environment.

**Note:** Based on this setting, the BigFix components can decide when it is safe

to assume the existence of newer functions across all the component in the

deployment. Individual agent interactions might be rejected if the interaction

does not comply with the limitations that are imposed by this setting.

The currently allowed values are:

v **0.0**, which means that no activity that is issued by BigFix Agents earlier than

V9.0, such as archive files and reports uploads, is prevented from running or

limited. This behavior applies also if the minimumSupportedClient service is

not set.

v **9.0**, which means that:

– Unsigned reports, such as the reports sent by BigFix Clients earlier than

V9.0, are discarded by FillDB.

– The upload of an unsigned archive file that is generated on a BigFix

Client earlier than V9.0, by an **archive now** command, for example, fails.

If you ran a fresh installation of BigFix V9.5.6 or later using a BES

Authorization file, by default all the BigFix Clients earlier than V9.0 are

prevented from joining your environment because the minimumSupportedClient

service is automatically set to **9.0**.

The value that is assigned to this service, if set, remains unchanged:

v If you upgraded to V9.5.6 or later.

v If you installed BigFix V9.5.6 or later using an existing masthead.

In both cases, if the service did not exist before, it will not exist afterward as

well.

The current value <VALUE> assigned in your environment to the

minimumSupportedClient service is displayed in the line x-bes-minimumsupported-

client-level: <VALUE> of the masthead file. You can see the current

value by running the following query on the BigFix Server from the BigFix

Query Application available on the BigFix WebUI:

Q: following text of last ": " of line whose (it starts with "x-bes-minimum-supported-client-level:" The syntax to run this service is:

./BESAdmin.sh -sitePvkLocation=<path+license.pvk> [-sitePvkPassword=<password>]

-minimumSupportedClient=<version>.<release>

If you omit to specify [sitePvkPassword=<password>], you are prompted to

enter the password interactively when the **BESAdmin.sh** runs.

For example, if you want to state that Agents earlier than V9.0 are not

supported in your BigFix environment, you can run the following command:

./BESAdmin.sh -sitePvkLocation=/license/license.pvk -minimumSupportedClient=9.0

Chapter 10. Installing on Linux systems **213**

**minimumSupportedRelay**

You can use this service, added with BigFix V9.5.6, to enforce specific criteria

that affect the BigFix Agent registration requests. If this service is enabled,

V9.5.6 Agents can continue to register to the V9.5.6 BigFix environment if their

registration requests are signed and sent across the Relays hierarchy using the

HTTPS protocol.

**Note:** Based on this service, the BigFix components can decide when it is safe

to enable newer functions across all the component in the deployment.

Individual agent interactions might be rejected if they do not comply with the

limitations that are imposed by this setting.

The currently allowed values are:

v **0.0.0**, which means that the BigFix Server accepts and manages:

– Signed and unsigned registration requests coming from BigFix Agents.

– Registration requests delivered from BigFix Agents using the HTTP or the

HTTPS protocols.

This behavior applies by default when you upgrade from previous versions

to BigFix V9.5.6 or later. In this case, the minimumSupportedRelay service is

not added automatically to your configuration during the upgrade.

v **9.5.6** or later, which means that:

– The BigFix Server enforces that registration requests coming from BigFix

Agents V9.5.6 or later must be properly signed.

– The BigFix Server and the Relays V9.5.6 or later enforce the use of the

HTTPS protocol when BigFix Agent registration data is exchanged.

Enforcing this behavior has the following side effects:

– BigFix Agents earlier than V9.0 cannot send registration requests to the

BigFix Server because they cannot communicate using the HTTPS

protocol.

– Because BigFix Relays with versions earlier than V9.5.6 cannot handle

correctly signed registration requests, any BigFix Client that uses those

Relays might be prevented from continuing to register, or might fall back

to a different parent Relay or directly to the Server.

If you ran a fresh installation of BigFix V9.5.6 or later using a License

Authorization file, be aware that the side effects that were just listed apply to

your BigFix deployment because, in this particular installation scenario, the

minimumSupportedRelay service is automatically set to **9.5.6** by default.

The current value <VALUE> assigned in your environment to the

minimumSupportedRelay service is displayed in the line x-bes-minimumsupported-

relay-level: <VALUE> of the masthead file. You can see the current

value by running the following query on the BigFix Server from the BigFix

Query Application available on the BigFix WebUI:

Q: following text of last ": " of line whose (it starts with

"x-bes-minimum-supported-relay-level:" ) of masthead of site "actionsite"

This query displays a value only when <VALUE> is set to **9.5.6**; if it is set to

**0.0.0**, it does not display a value.

The syntax to run this service is:

./BESAdmin.sh -sitePvkLocation=<path+license.pvk> [-sitePvkPassword=<password>]

-minimumSupportedRelay=<version>.<release>.<modification>

**214** IBM BigFix: Installation Guide

If you omit to specify [sitePvkPassword=<password>], you are prompted to

enter the password interactively when the **BESAdmin.sh** runs.

For example, if you want that only the registration requests that are signed and

carried through HTTPS are managed by your BigFix Server, you can run the

following command:

./BESAdmin.sh -sitePvkLocation=/license/license.pvk -minimumSupportedRelay=9.5.6

**propagateAllOperatorSites**

This service forces the server to propagate a new version of every operator site.

This command is useful after a server migration because you can be sure that

data is available for clients to gather and it prevents from failures. This is the

command syntax:

./BESAdmin.sh -propagateAllOperatorSites

**propertyidmapper**

This service creates, updates, and deletes a table (PropertyIDMap) in the

BFEnterprise database that maps retrieved property names for the SiteID,

AnalysisID, PropertyID used to reference properties in the

QUESTIONRESULTS and LONGQUESTIONRESULTS tables. It creates the

PropertyIDMap table if it does not exist (requires table creation permissions).

This service must be run to update the PropertyIDMap table after creating or

deleting a property.

The general syntax of this service is the following:

./BESAdmin.sh -propertyidmapper { -displaysettings | -run [property\_idmapper\_options]

| -schedule [property\_idmapper\_options] [scheduling options] }

The syntax of this service changes depending on the action you specify:

./BESAdmin.sh -propertyidmapper **-displaysettings**

./BESAdmin.sh -propertyidmapper **-run** [ -createtable ] [ -removetable ]

[ -lookupproperty=<propertyname> ]

./BESAdmin.sh -propertyidmapper **-schedule** [ -createtable -starttime=<yyyymmdd:hhmm>

[ -interval=<hours> ] | -disable ]

where:

v **displaysettings** shows the settings that are previously set with the schedule

action.

v **run** runs the tool with the specified settings. Before you use this option,

check the settings that affect the database by using the preview action.

v **schedule** schedules the tool to run at the specified time at each specified

interval. To disable the schedule action, use the -disable option.

For more information about the cleanup tasks log files, see “Logging Cleanup

Tasks Activities” on page 145.

**removecomputers**

The service runs database operations for the following sets of data:

v **Expired Computers** (-deleteexpiredcomputers) Marks computers as *deleted*

if they did not report in recently.

v **Deleted Computers** (-purgedeletedcomputers): Physically removes from the

database all te information about computers that are marked as *deleted* and

that did not report in for a long time.

v **Duplicate Computers** (-deleteduplicatedcomputers): Marks older

computers as deleted if a computer exists with the same computer name.

Chapter 10. Installing on Linux systems **215**

v **Removal of deleted Computers** (-removeDeletedComputers): Physically

removes computer data from the database for computers that are marked as

deleted since at least the indicated number of days (minimum 30).

v **Removal of uploaded Files** (-removeDeletedUploads): Physically removes

from the database the definition of uploaded files that are marked as

deleted.

v **Removal of uploaded files of removed computers**

(-eraseUploadFilesForRemovedComputers): Physically removes from the

BigFix server filesystem all files uploaded by clients whose definition has

been removed from the database.

v **Removal of Computers by name** (-removecomputersfile): Accepts a text file

with a list of computer names that are separated by new lines and removes

them from the deployment.

The general syntax of this service is:

./BESAdmin.sh -removecomputers { -displaysettings | -run [remove\_computers\_options]

| -schedule [remove\_computers\_options] [scheduling options]

| -preview [remove\_computers\_options] [preview options] }

Depending on the action that is specified, the syntax changes as follows:

./BESAdmin.sh -removecomputers **-displaysettings**

./BESAdmin.sh -removecomputers **-run** [ -deleteexpiredcomputers=<days> ]

[ -removeDeletedComputers=<days> ] [ -removeDeletedUploads ]

[ -eraseUploadFilesForRemovedComputers ]

[ -purgedeletedcomputers=<days> ]

[ -deleteduplicatedcomputers [ -duplicatedpropertyname=<propertyname> ] ]

[ -removecomputersfile=<path> ] [ -batchsize=<batch size> ]

./BESAdmin.sh -removecomputers **-schedule** [ [ -deleteexpiredcomputers=<days> ]

[ -removeDeletedComputers=<days> ] [ -removeDeletedUploads ]

[ -eraseUploadFilesForRemovedComputers ]

[ -purgedeletedcomputers=<days> ]

[ -deleteduplicatedcomputers [ -duplicatedpropertyname=<propertyname> ] ]

[ -removestarttime=<yyyymmdd:hhmm> [ -removeperiodicinterval=<hours> ] ]

[ -batchsize=<batch size> ] | -disable ]

./BESAdmin.sh -removecomputers **-preview** [ [ -deleteexpiredcomputers=<days> ]

[ -removeDeletedComputers=<days> ] [ -removeDeletedUploads ]

[ -eraseUploadFilesForRemovedComputers ]

[ -purgedeletedcomputers=<days> ][ -deleteduplicatedcomputers

[ -duplicatedpropertyname=<propertyname> ] ] | [ -scheduled ] ]

where:

v **displaysettings** shows the settings that are previously set with the schedule

action.

v **run** runs the tool with the specified settings. Before you use this option,

check the settings that affect the database by using the preview action.

v **schedule** schedules the tool to run at the specified time at each specified

interval. To disable the schedule action, use the -disable option.

v **preview** shows the number of database rows that are affected by the

specified settings. If no setting is passed to the preview option, the preview

performs the count by setting all options to true and using the default

values for dates. Use the -scheduled option to preview the scheduled

settings.

**Note:** When using option -removeDeletedComputers, the number of days must

be not less than 30.

**216** IBM BigFix: Installation Guide

For more information about the cleanup tasks log files, see “Logging Cleanup

Tasks Activities” on page 145.

**repair**

You can use this command to handle an inconsistency between the keys that

are stored in the database and the keys stored on the filesystem.

./BESAdmin.sh -repair -sitePvkLocation=<path+license.pvk>

[ -sitePvkPassword=<password> ]

If the keywords ServerSigningKey and ClientCAKey do not exist, they are

created under /var/opt/BESServer: This command also updates the licenses of

sites.

**reportencryption**

You can generate, rotate, enable, and disable encryption for report messaging

by running:

./BESAdmin.sh -reportencryption { -status |

-generatekey [-privateKeySize=<min|max>]

[-deploynow=yes | -deploynow=no -outkeypath=<path>]

-sitePvkLocation=<path+license.pvk> [-sitePvkPassword=<password>] |

-rotatekey [-privateKeySize=<min|max> ]

[-deploynow=yes | -deploynow=no -outkeypath=<path> ]

-sitePvkLocation=<path+license.pvk> [-sitePvkPassword=<password>] |

-enablekey -sitePvkLocation=<path+license.pvk> [-sitePvkPassword=<password>] |

-disablekey -sitePvkLocation=<path+license.pvk> [-sitePvkPassword=<password>] }

where:

**status** Shows the status of the encryption and which arguments you can use

for that status.

**generatekey**

Allows you to generate a new encryption key.

**rotatekey**

Allows you to change the encryption key.

**enablekey**

Allows you to enable the encryption key.

**disablekey**

Allows you to put the encryption key in PENDING state. If you run

again the reportencryption command with the disablekey argument,

the encryption changes from PENDING state to DISABLED.

**deploynow=yes**

Deploys the report encryption key to the server for decryption.

**deploynow=no -outkeypath=<path>**

The encryption key is not deployed to the server but it is saved in the

outkeypath path.

For more information about this command and its behavior, see Managing

Client Encryption.

**resetDatabaseEpoch**

To clear all console cache information in BigFix Enterprise Service V7.0 or later

versions. After running this command:

./BESAdmin.sh -resetDatabaseEpoch

subsequent console logins reload their cache files.

Chapter 10. Installing on Linux systems **217**

**resignsecuritydata**

If you get one of the following errors:

class SignedDataVerificationFailure

HTTP Error 18: An unknown error occurred while transferring data from the server

when you try to log in to the BigFix console, you must resign all the user

content in the database by entering the following command:

./BESAdmin.sh -resignSecurityData

This command resigns security data that uses the existing key file. You can

also specify the following parameter:

-mastheadLocation=<path+actionsite.afxm>

The complete syntax to run this service is:

./BESAdmin.sh -resignsecuritydata -sitePvkLocation=<path+license.pvk>

[ -sitePvkPassword=<password> ] -mastheadLocation=<path+actionsite.afxm>

**revokewebuicredentials**

You can revoke the authentication certificate of a specified WebUI instance.

The syntax to run this service is:

./BESAdmin.sh -revokewebuicredentials -hostname=<host> -sitePvkLocation=<path+license.pvk> -sitePvkPassword=<If an authentication certificate is issued for the specified hostname, this

certificate is revoked and the WebUI instance running on that hostname can no

longer connect to the root server.

**rotateserversigningkey**

You can rotate the server private key to have the key in the file system match

the key in the database. The command creates a new server signing key,

resigns all existing content that uses the new key, and revokes the old key.

The syntax to run this service is:

./BESAdmin.sh -rotateserversigningkey -sitePvkLocation=<path+license.pvk>

[ -sitePvkPassword=<password> ]

**revokewebuicredentials**

You can revoke the authentication certificate of a specified WebUI instance.

The syntax to run this service is:

./BESAdmin.sh /revokewebuicredentials /hostname=<host> /sitePvkLocation=<path+license.pvk> /sitePvkPassword=<If an authentication certificate is issued for the specified hostname, this

certificate is revoked and the WebUI instance running on that hostname can no

longer connect to the root server.

**securitysettings**

You can configure enhanced security options to follow the NIST security

standards by running the command:

./BESAdmin.sh -securitysettings -sitePvkLocation=<path+license.pvk>

[ -sitePvkPassword=<password> ]

{ -status | -enableEnhancedSecurity [-requireSHA256Downloads]

| -disableEnhancedSecurity | -requireSHA256Downloads

| -addDisabledTLSCiphers=<cipher1>,<cipher2>,... | -removeDisabledTLSCiphers=<cipher1>,<cipher2>,...

| -allowSHA1Downloads} }

where:

**218** IBM BigFix: Installation Guide

**status** Shows the status of the security settings set in your BigFix

environment.

Example:

./BESAdmin.sh -securitysettings -sitePvkLocation=/root/backup/license.pvk

-sitePvkPassword=mypassw0rd -status

Enhanced security is currently ENABLED

SHA-256 downloads are currently OPTIONAL

**enableEnhancedSecurity | disableEnhancedSecurity**

Enables or disables the enhanced security that adopts the SHA-256

cryptographic digest algorithm for all digital signatures and content

verification and the TLS 1.2 protocol for communications among the

Endpoint Manager components.

**Note:** If you use the **enableEnhancedSecurity** setting you break the

compatibility with an earlier version because BigFix version 9.0 or

earlier components cannot communicate with the BigFix version 9.5

server or relays. When you disable the enhanced security mode, the

BESRootServer service fails to restart automatically. To solve the

problem, restart the service manually.

**addDisabledTLSCiphers | removeDisabledTLSCiphers**

Starting from version 9.5.10, you can specify a comma-delimited list of

TLS ciphers suites that you want to disable by specifying the

**-addDisabledTLSCiphers** option. To re-enable any previously disabled

ciphers or ciphers suite, run the command using the

**-removeDisabledTLSCiphers** option. For example, you might decide

to disable specific TLS ciphers in your BigFix environment if you think

that they are compromised or questionable. After you ran the

command, propagate the change across the enterprise by manually

restarting the root server, the relays, the FillDB and the GatherDB

processes, and the Web Reports. The change is globally applied to your

environment if all the clients, the relays and the server are at version

9.5.10 or later. No change is applied to clients with versions earlier

than 9.5.10.

Each ciphers suite is identified by a cipher string. Click this link to see

the list of available OpenSSL cipher strings.

**Note:** You cannot disable the suites that are identified by the following

strings: ALL, HIGH, DEFAULT, aRSA, kRSA, RSA, AES,

COMPLIMENTOFALL, COMPLIMENTOFDEFAULT, TLSv1.2, FIPS.

If a ciphers suite that you want to disable is not found in current

version of OpenSSL, the **BESAdmin** command adds the cipher string

to the list of disabled cyphers anyway, and issues a warning message.

For example:

./BESAdmin.sh -sitePvkLocation=<path+license.pvk> -sitePvkPassword=<password> -securitysettings

-addDisabledTLSCiphers=3DES,RC4,NotInOpenSSL

3DES is found in libBEScrypto - OpenSSL 1.0.2j-fips 26 Sep 2016

Warning: NotInOpenSSL is not found in libBEScrypto - OpenSSL 1.0.2j-fips 26 Sep 2016

RC4 is found in libBEScrypto - OpenSSL 1.0.2j-fips 26 Sep 2016

Updating masthead

Restarting stopped services

Disabled TLS cipher list changed. Server and relay services need to be restarted after Enhanced security is currently DISABLED

SHA-256 downloads are currently OPTIONAL

Ciphers disabled for TLS connections: 3DES NotInOpenSSL RC4

Chapter 10. Installing on Linux systems **219**

Similarly, if a cipher string that you want to re-enable does not appear

in the list of disabled ciphers, the **BESAdmin** command displays an

information message and ignores the request for that specific string.

For example:

./BESAdmin.sh -sitePvkLocation=<path+license.pvk> -sitePvkPassword=<password> -securitysettings

-removeDisabledTLSCiphers=3DES,NotInMasthead

TLS cipher not found in the disabled cipher list: NotInMasthead

Updating masthead

Restarting stopped services

Disabled TLS cipher list changed. Server and relay services need to be restarted after masthead Enhanced security is currently DISABLED

SHA-256 downloads are currently OPTIONAL

Ciphers disabled for TLS connections: NotInOpenSSL RC4

If you want to see the current list of disabled ciphers, run the

command as follows:

./BESAdmin.sh -sitePvkLocation=<path+license.pvk> -sitePvkPassword=<password> -securitysettings

Enhanced security is currently DISABLED

SHA-256 downloads are currently OPTIONAL

Ciphers disabled for TLS connections: NotInOpenSSL RC4

**requireSHA256Downloads**

Ensures that data has not changed after you download it using the

SHA-256 algorithm.

**Note:** The **Require SHA-256 Downloads** option is available only if

you selected to **Enable Enhanced Security**.

**allowSHA1Downloads**

Ensures that the file download integrity check is run using the SHA-1

algorithm.

For more information about the BigFix Enhanced Security feature and the

supported security configuration, see Chapter 5, “Security Configuration

Scenarios,” on page 37.

**setadvancedoptions**

You can list or configure any global settings that apply to your particular

installation. The complete syntax to run this service is:

./BESAdmin.sh -setadvancedoptions -sitePvkLocation=<path+license.pvk>

[-sitePvkPassword=<password>]

{ -list | -display

| [ -f ] -delete option\_name

| [ -f ] -update option\_name=option\_value }

For example:

v To customize the Console or Web Report login banner, enter following

command:

./BESAdmin.sh -setadvancedoptions -sitePvkLocation=/root/backup/license.pvk

-sitePvkPassword=pippo000 -update loginWarningBanner=’new message’

v To ensure that the BigFix environment includes only the clients that have all

the relays in the registration chain upgraded to the same version of product

that is installed on the server, enter the following command:

./BESAdmin.sh -setadvancedoptions -sitePvkLocation=/root/backup/license.pvk

-sitePvkPassword=pippo000 -update requireSignedRegistration=true

**220** IBM BigFix: Installation Guide

v To ensure that the BigFix server checks for the existence of the client

signature in the registration request before adding the client to the BigFix

network, run the following command:

./BESAdmin.sh -setadvancedoptions -sitePvkLocation=/root/backup/license.pvk

-sitePvkPassword=pippo000 -update requireSignedRegistration=true

v If your BigFix Server is V9.5.7 or later, to avoid having duplicate computer

entries when the endpoints are detected as possible clones by the Server, run

the following command:

./BESAdmin.sh -setadvancedoptions -sitePvkLocation=/root/backup/license.pvk

-sitePvkPassword=pippo000 -update clientIdentityMatch=100

For a list of available options that you can set, see “List of advanced options”

on page 127.

**setproxy**

If your enterprise uses a proxy to access the Internet, you must set a proxy

connection to enable the BigFix server to gather content from sites and to do

component-to-component communication or to download files.

For more information about how to run the command and about the values to

use for each argument, see “Setting a proxy connection on the server” on page

274.

**syncmastheadandlicense**

When you upgrade the product, you must use this option to synchronize the

update license with the masthead and resign all content in the database with

SHA-256. The syntax to run this service is:

./BESAdmin.sh -syncmastheadandlicense -sitePvkLocation=<path+license.pvk>

[-sitePvkPassword=<password>]

**updatepassword**

You can modify the password that is used for authentication by product

components in specific configurations.

The syntax to run this service is:

./BESAdmin.sh -updatepassword -type=<server\_db|dsa\_db>

[-password=<password>] -sitePvkLocation=<path+license.pvk>

[-sitePvkPassword=<pvk\_password>]

where:

**-type=server\_db**

Specify this value to update the password that is used by the server to

authenticate with the database.

**-type=dsa\_db**

Specify this value to update the password that is used in a DSA

configuration by a server to authenticate with the database.

The settings -password and -sitePvkPassword are optional, if they are not

specified in the command syntax their value is requested interactively at run

time. The password set by this command is obfuscated.