Name

create-virtual-server– creates the named virtual server

Synopsis

```
create-virtual-server
[--terse={true|false}][ --echo={true|false} ]
[ --interactive={true|false} ] [ --host  host]
[--port port] [--secure| -s ] [ --user  admin_user]
[--passwordfile filename] [--help]
[ --target  target]
--hosts hosts [--networklisteners network-listeners]
[--defaultwebmodule default_web_module]
[--state state] [--logfile log_file]
[--property (name=value)[:name=value]*]
virtual_server_id
```

Description

The `create-virtual-server` command creates the named virtual server. Virtualization in the Application Server allows multiple URL domains to be served by a single HTTP server process that is listening on multiple host addresses. If the application is available at two virtual servers, they still share the same physical resource pools.

This command is supported in remote mode only.

Options

`-t --terse`
Indicates that any output data must be very concise, typically avoiding human-friendly sentences and favoring well-formatted data for consumption by a script. Default is false.

`-e --echo`
If set to true, the command-line statement is echoed on the standard output. Default is false.

`-I --interactive`
If set to true (default), only the required password options are prompted.

`-H --host`
The machine name where the domain administration server is running. The default value is `localhost`.

`-p --port`
The HTTP port or HTTPS port for administration. This port is the port in the URL that you specify in your web browser to manage the domain, for example, `http://localhost:4848`.

The default port number for administration is 4848.

`-s --secure`
If set to true, uses SSL/TLS to communicate with the domain administration server.

The default is false.

`-u --user`
The user name of the authorized administrative user of the domain administration server.

If you have authenticated to a domain by using the `asadmin login` command, you need not specify the `--user` option for subsequent operations on the domain.

`--passwordfile`
Specifies the name, including the full path, of a file that contains the password entries in a specific format.

The entry for a password must have the `AS_ADMIN_` prefix followed by the password name in uppercase letters. For example, to specify the password for the domain administration server, use an entry with the following format:

`AS_ADMIN_PASSWORD=`*password*

In this example, *password* is the actual administrator password.

The following other passwords can also be specified:

- `AS_ADMIN_MAPPEDPASSWORD`
- `AS_ADMIN_USERPASSWORD`
- `AS_ADMIN_ALIASPASSWORD`

All remote commands must specify the administration password to authenticate to the domain administration server. The password can be specified by one of the following means:

- Through the `--passwordfile` option
- Through the `asadmin login` command
- Interactively at the command prompt

The `asadmin login` command can be used only to specify the administration password. For other passwords that remote commands require, use the `--passwordfile` option or specify them at the command prompt.

After authenticating to a domain by using the `asadmin login` command, you need not specify the administration password through the `--passwordfile` option for subsequent operations on the domain. However, only the `AS_ADMIN_PASSWORD` option is not required. You still must provide the other passwords, for example, `AS_ADMIN_USERPASSWORD`, when required by individual commands, such as `update-file-user`.

For security reasons, a password that is specified as an environment variable is not read by the `asadmin` command.

The default value for `AS_ADMIN_MASTERPASSWORD` is `changeit`.

`--help`
Displays the help text for the command.

`--target`
Do not specify this option. This option is retained for compatibility with other releases. If you specify this option, a syntax error does not occur. Instead, the command runs successfully and the option is silently ignored.

`--hosts`
A comma-separated (,) list of values allowed in the host request header to select the current virtual server. Each virtual server that is configured to the same connection group must have a unique host for that group.

`--networklisteners`
A comma-separated (,) list of network listener IDs. Required only for a virtual server that is not the default virtual server.

`--defaultwebmodule`
The standalone web module associated with this virtual server by default.

`--state`
Determines whether a virtual server is active (on) or inactive (off or disabled). Default is active (on). When inactive, the virtual server does not service requests.

`--logfile`
Name of the file where log entries for this virtual server are to be written. By default, this is the server log.

`--property`
Optional attribute name/value pairs for configuring the virtual server. The following properties are available:

`docroot`
Absolute path to root document directory for server.

`accesslog`
Absolute path to server access logs.

`sso-enabled`
If false, single sign-on is disabled for this virtual server, and users must authenticate separately to every application on the virtual server. Single sign-on across applications on the Application Server is supported by servlets and JSP pages. This feature allows multiple applications that require the same user sign-on information to share this information, rather than have the user sign on separately for each application. The default value is true.

`sso-max-inactive-seconds`
Specifies the number of seconds after which a user's single sign-on record becomes eligible for purging if no client activity is received. Since single sign-on applies across several applications on the same virtual server, access to any of the applications keeps the single sign-on record active. The default value is 300 seconds (5 minutes). Higher values provide longer single sign-on persistence for users, but at the expense of more memory use on the server.

`sso-reap-interval-seconds`
Specifies the number of seconds between purges of expired single sign-on records. The default value is 60.

`default-web-xml`
Indicates the location of the file `default-web.xml`. The default location is `$[S1AS_HOME]/domains/domain1/config/default–web.xml`.

`allowLinking`
If the value of this property is true, resources that are symbolic links will be served for all web applications deployed on this virtual server. Individual web applications may override this setting by using the property `allowLinking` under the `sun-web-app` element in the `sun-web.xml` file:

```
<sun-web-app>
<property name=”allowLinking” value=”[true|false]”/>
</sun-web-app>
```

The default value is true.

`accessLogWriteInterval`
Indicates the number of seconds before the log will be written to the disk. The access log is written when the buffer is full or when the interval expires. If the value is 0 (zero), then the buffer is always written even if it is not full. This means that each time the server is accessed, the log message is stored directly to the file.

`accessLogBufferSize`
Specifies the size, in bytes, of the buffer where access log calls are stored.

`allowRemoteAddress`
This is a comma-separated list of regular expression patterns to which the remote client's IP address is compared. If this property is specified, the remote address must match for this request to be accepted. If this property is not specified, all requests will be accepted unless the remote address matches a `denyRemoteAddress` pattern. The default value for this property is null.

`denyRemoteAddress`
This is a comma-separated list of regular expression patterns to which the remote client's IP address is compared. If this property is specified, the remote address must not match for this request to be accepted. If this property is not specified, request acceptance is governed solely by the `allowRemoteAddress` property. The default value for this property is null.

`allowRemoteHost`
This is a comma-separated list of regular expression patterns to which the remote client's host name (as returned by `java.net.Socket.getInetAddress().getHostName()`) is compared. If this property is specified, the remote host name must match for this request to be accepted. If this property is not specified, all requests will be accepted unless the remote host name matches a denyRemoteHost pattern. The default value for this property is null.

`denyRemoteHost`
This is a comma-separated list of regular expression patterns to which the remote client's host name (as returned by `java.net.Socket.getInetAddress().getHostName()`) is compared. If this property is specified, the remote host name must not match for this request to be accepted. If this property is not specified, request acceptance is governed solely by the `allowRemoteHost` property. The default value for this property is null.

Operands

virtual_server_id
Identifies the unique ID for the virtual server to be created. This ID cannot begin with a number.

Examples

EXAMPLE 1 Using the create-virtual-server command

The following command creates a virtual server named `sampleServer`:

```
asadmin> create-virtual-server --user admin1
--passwordfile passwords.txt --hosts pigeon,localhost sampleServer
Command create-virtual-server executed successfully.
```

Exit Status

0	command executed successfully
1	error in executing the command

See Also

`delete-virtual-server`(1), `list-virtual-servers`(1), `create-http-listener`(1)