SmartLink configuration

DME Server 3.5

Document version:	1.3
Date:	2010-03-22
Circulation/Restrictions:	Internal/Excitor partners
Applies to:	DME Server 3.5

Table of contents

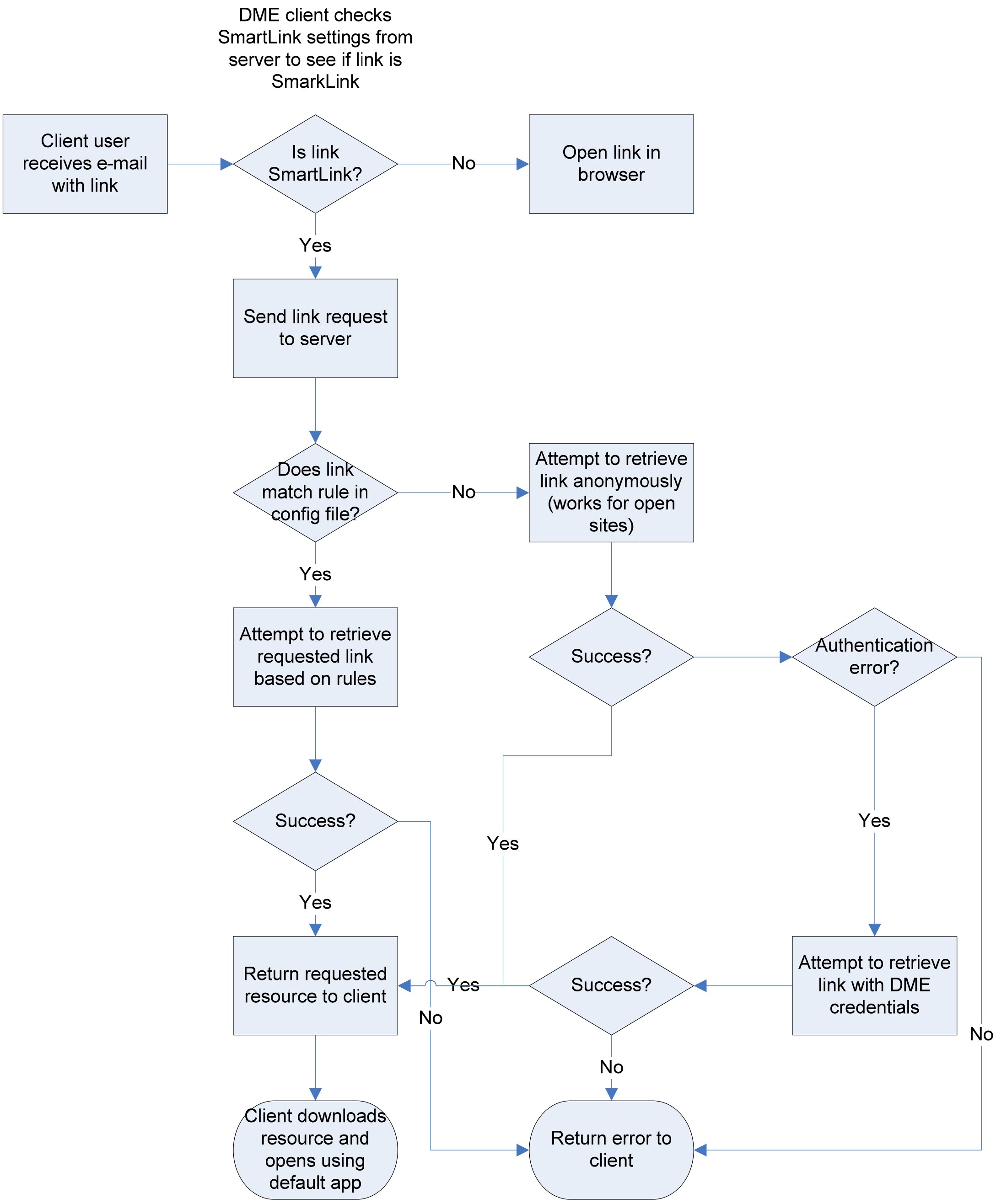
SmartLink configuration

DME SmartLink allows integration to internal, web-based document management systems. With DME SmartLink, users can access internal documents directly via the DME client when not on the internal networks.

As an add-on module introduced in DME 3.0, SmartLink allows employees to access web links (URLs) to documents directly from within the body of an e-mail. In order to avoid exposing the systems to the entire Internet, DME proxies these requests, authenticates as the user, retrieves the documents and presents them to the user.

How it works

The way SmartLink works can be illustrated as in the following diagram:


DME client checks SmartLink settings from server to see if link is SmarkLink
Client user receives e-mail with link
Is link SmartLink?
No
Open link in browser
Yes
Send link request to server
Does link match rule in config file?
No
Attempt to retrieve link anonymously (works for open sites)
Yes
Attempt to retrieve requested link based on rules
Success?
Authentication error?
Success?
Yes
Yes
Yes
Return requested resource to client
Yes
No
Success?
Attempt to retrieve link with DME credentials
No
No
Client downloads resource and opens using default app
Return error to client

When the DME client parses an open e-mail for web links in order to make them "active", it matches each link with a list of smart links. This list is retrieved from the server as part of the system information settings, and is maintained in the **SmartLink** group of settings on the server. The list could for instance look like this:

```
https://intranet.company.com|http://public.company.com
```

With this list, the DME client would for instance recognize the link

```
https://intranet.company.com/Docs/Documents/3429459435/brochure.doc
```

as a SmartLink. When the user clicks the link, the client will attempt to send a download request for the document from the intranet portal to the DME server, rather than attempting to display it in a browser (which would fail if the device is not connected to the internal LAN by VPN).

The DME connector set up to provide SmartLinks checks the resource request from the client against a SmartLink configuration file. Based on rules in this file, the server will act as a proxy and authenticate against the server which contains the resource, using standard HTTP authentication mechanisms (Basic, NTLM, or Digest Authentication) or Form-Based Authentication (FBA). DME will pass the user's credentials to the resource server.

If successful in logging in and finding the resource, the DME server presents the resource to the client, which in turn downloads it to the encrypted DME area, and opens it using the default application – for instance Pocket Word or QuickOffice. In case of failure, the server sends an error message to the client instead.

If a link is recognized as a SmartLink, but no rules or configuration have been set up for the site in question, DME first inspects the response from the site before attempting any authentication. If the request requires authentication, DME will attempt an authentication based on the username and password entered on the device in the following order: For secure sites (HTTPS): Basic, NTLM, Digest. For regular HTTP sites: NTLM, Digest, Basic. This is the default configuration.

Setting up the SmartLink configuration file

The setup of SmartLink client match strings is described in the DME Server Administration Guide (http://documentation.excitor.com/server/3_5/index.htm?topic=3896).

The SmartLink configuration file is an XML file called `smartlinkConfig.xml`, which should be located in the `conf` directory of the connector responsible for providing the SmartLink service (see the online help for the **Connector setup** > **Functions** page in the DME Server Administration Web Interface).

On *Linux*, this is at `/var/dme/instances/base/connector/conf` (where `base` is the default DME instance på Linux). On *Windows*, the path is `C:\program files\dmeconnector\conf`.
If SmartLink is enabled on multiple connectors, each connector must have a copy of the SmartLink configuration file.

Apart from the xml root tag (`smartlinkConfig`), the file contains one `<sites>` tag, which may contain any number of `<site>` tags. Each site is defined within a set of `<site>` tags.

The definition of a site has the form `<sitename>` followed by either `<AuthConfig>` or `<FbAuthConfig>`.

Site specification

The `<site>` specification consists of the following tags. Note that unless otherwise specified, all tags are parenthetical – that is, they must be closed by a corresponding `</ >` tag.

`<sitename>` *Mandatory* *Solitary*	This is the name of the site on which the resource should be found. This must match one of the sites listed in **Device settings** on the DME server. *Example:* `<sitename> https://sharepoint.company.com</sitename>`
`<AuthConfig>` ***Optional*** ***Solitary***	This tag defines protocol-based access to the site. You must use this or the `<FBAuthConfig>` tag.

`<FbAuthConfig>` ***Optional*** ***Solitary***	This tag defines form-based authentication access to the site. You must use this or the `<AuthConfig>` tag.

Specification of HTTP protocol-based authentication

DME supports the following authentication protocols: Basic, NTLM, and Digest. For detailed information about these protocols, see http://en.wikipedia.org/wiki/Basic_authentication, http://en.wikipedia.org/wiki/NTLM, and http://en.wikipedia.org/wiki/Digest_authentication, respectively.

The following tags apply to protocol-based authentication.

`<username>` *Optional, Solitary*	In this tag you can specify a username used for the authentication. Enter a username in this tag if all DME users should have the same access to the current site. If you leave the tag empty, DME will use the credentials sent in from the device. *Example:* `<username>exampleuser</username>`
`<password>` *Optional, Solitary*	In this tag you can specify the password of the user specified in the username tag. If you leave the tag empty, DME will use the credentials sent in from the device.
`<domain>` *Optional, Solitary*	In this tag you can specify a domain, if the site requires an AD domain. If you are using NTLM authentication, and you do not specify a domain in this tag, DME will attempt to extract a domain from the username from the device (anything after a "@" in the username).

Specification of form-based authentication

The tags used for protocol-based authentication also apply here: `<username>`, `<password>`, and `<domain>`. Apart from these, you can specify the following tags:

`<postUrl>` *Mandatory* *Solitary*	This is the URL to which DME will post the login request when the site specified in `<sitename>` is matched. *Example:* `<postUrl> https://sharepoint.company.com/login.aspx</postUrl>`
`<usernameField>` *Mandatory* *Solitary*	This is the HTML `name` of the field that holds the username on the authentication page to which the `<postUrl>` tag refers. The username will be entered in this field. If the website requires special encoding, you can add that information as an `encoding` attribute. If `encoding` is not specified, `utf-8` is used by default. Only specify this if the default does not work. *Example:* `<usernameField encoding="ISO-8859-1">sp_user</usernameField>`
`<passwordField>` *Mandatory* *Solitary*	This is the HTML `name` of the field that holds the password on the authentication page to which the `<postUrl>` tag refers. The password will be entered in this field. See `usernameField` above for a description of the `encoding` attribute. *Example:* `<passwordField encoding="ISO-8859-1">sp_pw</passwordField>`

Examples

The following are short examples of different SmartLink configurations.

Form-based authentication

The following example logs the user on to the Excitor Partner site. The example posts the default credentials (username and password used on the device) to the fields **`Username`** and **`Password`** at **`<postUrl>`**. The post is successful if a header called **`Set-Cookie`** is returned with a value starting with **`DW_Extranet`**.

```
<sites>
    <site>
        <sitename>http://www.excitor.com</sitename>
        <FbAuthConfig>
            <postUrl>
                <![CDATA[http://www.excitor.com/Default.aspx?ID=78&Purge=True]]>
            </postUrl>
            <usernameField encoding="ISO-8859-1">Username</usernameField>
            <passwordField encoding="ISO-8859-1">Password</passwordField>
        </FbAuthConfig>
    </site>
</sites>
```

Note that if a string contains characters that have special meaning in XML, you can always wrap the string in **`<![CDATA[ … ]]>`**. This applies to the **`postUrl`** string above, which contains an "**&**". This is standard functionality in XML parsers.

Protocol-based authentication

The following example attempts to log a user on to the local Sharepoint site, which is located at **`https://sharepoint`**. For credentials, the username from the device is used, and the domain name is hardcoded (**`excitor.local`**). Basic authentication is attempted first, then NTLM.

```
<sites>
    <site>
        <sitename>https://sharepoint</sitename>
        <AuthConfig>
            <domain>excitor.local</domain>
            <authTypes>Basic, NTLM</authTypes>
        </AuthConfig>
    </site>
</sites>
```