

# Coveo Enterprise Search

## Deploying on Computers without a Domain

Applies to CES 4 and later

*Windows NT* domains provide security and authentication services that are very useful in everyday life and *Coveo Enterprise Search* (CES) relies heavily on them. Still, there are a few situations where a user might want to run CES without having his computer hooked on a domain, such as a Web site demonstration on a laptop computer or someone who works at home and does not have access to the company's network. The following presents the limitations of such an environment and how they can affect CES installations.

### Why to Use Windows NT Domains

CES uses NT domains for authentication purposes. They are required to manage security-related interactions between the server hosting the CES installation and the numerous computers that it accesses to index content. They are also required for CES to authenticate the clients performing queries and thus restrict them to documents that they have access to.

Under *Windows NT*, all processes (for instance word processors, browsers, the CES service, etc.) run under a specific user account. Usually, it is the account of the user that started those processes, but for services like CES, the user account is specified in the *Windows Control Panel > Administrative Tools > Services > MMC* snap-in. The local and remote computers use this user account to determine what the process can do and what it cannot.

When a process attempts to access a resource on another computer (on the associated user's behalf), both computers use the services provided by the NT domain to pass on credentials from one computer to the other. The source computer employs user credentials originating from the NT domain that the target computer trusts. This permits a user or process to access computers without having to explicitly log on to them, because the user or process can use the same credentials on all computers of the NT domain.

When a computer runs in a domain-less environment, it cannot use credentials that come from another computer. All remote users that attempt to access resources on that computer will have to provide complete logon information (name and password) to be authenticated on the target computer because no trusted central authority can provide cross-computer credentials management.

There is an exception to this rule. If a user called *John* attempts to connect to computer B from computer A in a domain-less environment, and that computer B also defines a user named *John* who has the same password as user *John* on computer A, computer B will automatically map John's connection to its local user *John* and give him access according to these credentials. This feature is useful when one needs to work with several computers using a similar user account, but can become confusing as it may bring someone to think that local credentials can travel over the network without the domain, whereas they are just mapped to similar credentials onto remote computers.

### Coveo Enterprise Search and Domain-less Environments

Very often, CES is configured to index documents that reside on remote computers, using various protocols like HTTP, file sharing, etc. When accessing remote sources that require specific credentials, CES uses the user account under which the service process runs. In a domain environment, credentials are safely passed on to computers and everything works as expected. But if no domain is available, the credentials are not transmitted and no data is indexed.

A similar problem happens when a client attempts to connect to CES from another computer in a domain-less environment. When IIS (the Web server) receives a query, it attempts to retrieve credentials for the user that sent the query. If the credentials cannot be sent, it will reply with an *access denied* status code. And then, a dialog box prompts the user for a username and a password. If the Web server accepts them, the query will be processed; however, if the user cannot provide a valid username and password (recognized by the server), the connection will completely fail.

Therefore, CES can work correctly in a domain-less environment but it will fail to index any remote source (except those that use the HTTP protocol, since it is possible to specify a name and password for each source). Also, users will be able to create queries from the local computer directly and remote computers, if they provide a username and password when they connect to the search interface.

### Document Level Security and Domain-less Environments

CES provides *document level security*, in other words the integration of file access permissions. This feature allows the security settings of the documents that are crawled on file or MAPI servers to be indexed and those are used afterwards to restrict the results that users get when they perform queries. In a completely domain-less environment, this feature will only be of use when indexing files located on the local computer, since the only files that can be accessed on remote computers are those that aren't restricted in any way. However, when the computer (such as a laptop computer) that hosts CES is only occasionally on a domain, the index could contain files that can be accessed by domain users. When the domain is not available, those files may in turn become unavailable because no local user of the computer that runs CES can have permissions on a remote server. Still, if the user that performs the query is logged on using a domain account that was cached on the CES computer, he may still have access to the files, because CES will be able to retrieve its domain group memberships from the cached information, and match them against the one that he indexed.

### Conclusion

In brief, although CES can operate in a domain-less environment, some restrictions apply. Problems may occur when indexing files that are located on remote computers, but they are all consequences of the limitations of the environment itself, and not of the CES design. Users who want to use CES in a domain-less environment will find that it is perfectly possible to do so, unless they have advanced needs for security features like file level security and user authentication.