Adding Data from Logstash (HTTP/S)

Logstash XpoLog Integration

XpoLog's architecture allows receiving data sent by logstash from JSON data objects. In order to do this, a HTTP/S listener account should be created in XpoLog for receiving the data on a specific network port, and the XpoLog logstash output should be sending data to this listener.

Technical Details

The logstash configuration file should look like the following:

```
input {

        file {

        # The path to the files to be sent to XpoLog
        path => "FULL_PATH_TO_LOGS_DIRECTORY/FILE"
        # The default type that will be sent to XpoLog - used to name the log(s) that will be created in XpoLog. If specific xpologType (see below) is
        sent it will override this general type.
        type => "DEFAULT_TYPE_OF_LOGS_IN_XPOLOG"
        start_position => "beginning"
        sincedb_path => "SINCEDB_PATH"

        }

}

filter {

                grok {

                        match => ["path","%{GREEDYDATA}/%{DATA:parent}/%{GREEDYDATA}\.log"]
                        # Adding 3 fields to the JSON (optional).
                        # xpologPath - '->' separated list of folders structure to place the log(s) in XpoLog Folders and Logs
                        add_field => [ "xpologPath", "FOLDER_AND_LOGS_PATH_IN_XPOLOG" ]
                        # xpologName - the name of the log that will be created in XpoLog
                        add_field => [ "xpologName", "DEFAULT_NAME_OF_THE_LOG_IN_XPOLOG" ]
                        # xpologType - the type of the log in XpoLog. XpoLog will automatically look for templates with the exact same type and, if
                        found, will apply the template's pattern on the received log
                        add_field => [ "xpologType", "DEFAULT_TYPE_OF_THE_LOG_IN_XPOLOG" ]

                }

                if [path] =~ "FILE_NAME_1" {

                        # Modifying the parameters for specific file FILE_NAME_1 in the directory FULL_PATH_TO_LOGS_DIRECTORY
                        mutate { replace => { xpologPath => "FOLDER_1->FOLDER_2" } }
                        mutate { replace => { xpologType => "TYPE_1" } }
                        mutate { replace => { xpologName => "NAME_1" } }

                } else if [path] =~ "FILE_NAME_2" {

                        # Modifying the parameters for specific file FILE_NAME_2 in the directory FULL_PATH_TO_LOGS_DIRECTORY
                        mutate { replace => { xpologPath => "FOLDER_3->FOLDER_4" } }
                        mutate { replace => { xpologType => "TYPE_2" } }
                        mutate { replace => { xpologName => "NAME_2" } }

                }

}

output {

        http {

                format => "json"
                http_method => "post"
                url => "URL_COPIED_FROM_XPOLOG_LISTENER_INCLD_TOKEN"
```

```
        }
    }
```