2026

GenericSSO

Enterprise Single Sign-On (SSO) Module

Technical Reference Guide

Version: 1.0
Date: 16th Feb 2026

1. Introduction

1.1 Purpose

This document provides the technical reference for the **GenericSSO Enterprise Single Sign-On Module**, including supported commands, configuration standards, implementation syntax, and operational guidelines.

1.2 Scope

This guide is intended for:

- System Administrators
- Application Support Teams
- Automation Engineers
- Security Teams

---

2. System Requirements

- Minimum Requirement: **.NET Framework 4.8**
- Windows Operating System
- Configurable file: data.ini

---

3. Configuration Standards

3.1 Application Section Requirement

All parameters must begin with the **Application Name**, defined as a [SECTION] in the data.ini file.

3.2 Case Sensitivity

All commands in data.ini are **case-sensitive**.
Incorrect casing may result in execution failure.

3.3 Unsupported Special Characters

The following characters are not supported:

- Backslash (\)
- Double Quotes (")

---

4. Supported Commands Overview

Command	Description
keys	Executes SendKeys Send
keyswait	Executes SendKeys SendWait
staticpassword	Uses encrypted password file
link	Clicks hyperlink (full text match)
partial-link	Clicks hyperlink (partial match)
button	Automates button interaction
field1..fieldN	Handles dynamic input parameters
msgbox	Displays message dialog
sleep	Pauses execution
blockinput	Enables/Disables user input
bringtofront	Brings process to foreground
checkfname	Validates focused element name
checkfcname	Validates focused element class name
checkfaid	Validates focused element Automation ID
while	Waits for validation condition

---

5. Command Specifications

5.1 keys
Executes Microsoft SendKeys Send method.
Example: keys:{TAB};keys:{ENTER}

5.2 keyswait
Executes Microsoft SendKeys SendWait method.
Example: keyswait:{TAB};keyswait:{ENTER}

5.3 staticpassword
Retrieves encrypted password from file.
Syntax: staticpassword:filename:id/name/xpath/tagname/classname/send/sendwait

5.4 link
Selects hyperlink matching full text.
Example: link:Continue to this Website

5.5 partial-link
Selects hyperlink using partial text match.
Example: partial-link:Continue

5.6 button

Automates button interaction or focuses UI element.

Supported Attributes:

- id
- name
- xpath
- classname
- tagname

Example: button:id:userName;button:name:Password

5.7 field1 … fieldN

Processes dynamic parameters passed after Application Name.

Sequentially reads:

- field1
- field2
- field3
- …

If used with send, the placeholder value is transmitted directly.

Example: field1:xpath://*[@id="email"];field2:send

5.8 msgbox

Displays message dialog defined in data.ini.

Example: msgbox:Process initiated;msgbox:Please wait

5.9 sleep

Pauses execution for specified duration (milliseconds).

Example: sleep:1000

5.10 blockinput

Blocks or enables user input during automation.

Recommended:

- Enable at start
- Disable at completion

Example: blockinput:true;blockinput:false

5.11 bringtofront

Brings process defined under PROCESSNAME in data.ini to foreground.

Example: bringtofront

5.12 checkfname

Validates focused element Name attribute.
Example: checkfname:username;checkfname:password

5.13 checkfcname
Validates focused element Class Name attribute.
Example: checkfcname:username; checkfcname:password

5.14 checkfaid
Validates focused element Automation ID.
Example: checkfaid:10001;checkfaid:10002

5.15 while
Waits until validation condition is satisfied.
Supported Extensions:

- checkfname
- checkfcname
- checkfaid

Example: while:checkfname:username;;while:checkfcname:Password

---

6. Execution Flow Best Practices

1. Define Application Name section in data.ini
2. Maintain proper command sequence
3. Follow case-sensitive syntax
4. Avoid unsupported characters
5. Validate UI focus before sending credentials
6. Disable blockinput at automation completion

---

7. Security Considerations

- Use staticpassword for encrypted credentials.
- Never store plain-text passwords.
- Restrict access to data.ini.
- Validate element focus before transmitting sensitive information.
- Limit file access permissions to authorized personnel only.
