Utility - CryptSharp

User Guide

Document Revision 1.0

Trademarks and copyrights

The descriptions and screenshots contained in this document are licensed under the Creative Commons Attribution-ShareAlike (CC-BY-SA) 3.0 license <https://creativecommons.org/licenses/by-sa/3.0/>.

**© Blue Prism Limited, 2001 – 2020**

®“Blue Prism”, the “Blue Prism” logo and Prism device are either trademarks or registered trademarks of Blue Prism Limited and its affiliates. All Rights Reserved

All trademarks are hereby acknowledged and are used to the benefit of their respective owners.  
Blue Prism is not responsible for the content of external websites referenced by this document.

Blue Prism Limited, 2 Cinnamon Park, Crab Lane, Warrington, WA2 0XP, United Kingdom  
Registered in England: Reg. No. 4260035. Tel: +44 870 879 3000. Web: [www.blueprism.com](file:///C:\Users\adutton\Documents\Rebranding\Templates\www.blueprism.com)

Contents

[1. Introduction 4](#_Toc47708857)

[2. Prerequisites 5](#_Toc47708858)

[3. Configuration 6](#_Toc47708859)

[4. Using the Asset 8](#_Toc47708860)

[4.1 Blowfish 8](#_Toc47708861)

[4.2 Blowfish Verify 8](#_Toc47708862)

[4.3 ExtendedDESEncrypt 9](#_Toc47708863)

[4.4 ExtendedDESEncrypt Verify 9](#_Toc47708864)

[4.5 LDAPEncrypt 10](#_Toc47708865)

[4.6 LDAPEncrypt Verify 10](#_Toc47708866)

[4.7 MD5Encrypt 11](#_Toc47708867)

[4.8 MD5Encrypt Verify 11](#_Toc47708868)

[4.9 PhpassEncrypt 12](#_Toc47708869)

[4.10 PhpassEncrypt Verify 12](#_Toc47708870)

[4.11 SHA256Encrypt 13](#_Toc47708871)

[4.12 SHA256Encrypt Verify 13](#_Toc47708872)

[4.13 SHA512Encrypt 14](#_Toc47708873)

[4.14 SHA512Encrypt Verify 14](#_Toc47708874)

[4.15 DESEncrypt 15](#_Toc47708875)

[4.16 DESEncrypt Verify 15](#_Toc47708876)

[5. Support 16](#_Toc47708877)

# Introduction

CryptSharp is a C# library created by James F Bellinger. It provides a number of password algorithms. This VBO is a wrapper around that library to present the user with the ability to create and verify password hashes using these algorithms.

The options available are:

* BCrypt (Blowfish)
* LDAP
* MD5
* PHPass (Wordpress, phpBB, Drupal)
* SHA256/512
* Traditional and Extended DES.

# Prerequisites

List any prerequisites necessary for the proper functioning of the Asset. Examples include:

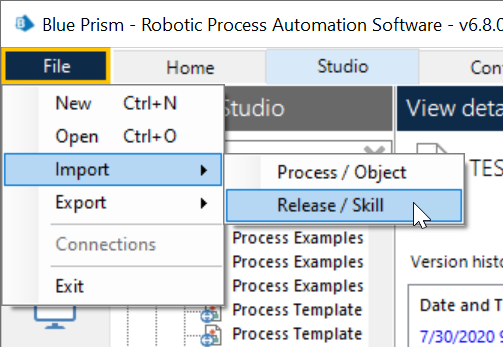
* Blue Prism V6.8 or later.
* CryptSharp dll (available from <https://www.nuget.org/packages/CryptSharpOfficial/>)
* The Utility – CryptSharp bprelease availabl at <https://github.com/blue-prism/CryptSharp>

# Configuration

The asset can be downloaded from the Digital Exchange (DX) at or from Github at <https://github.com/blue-prism/CryptSharp>

The bprelease file contains the Utility CryptSharp VBO and also a process example of its use. The file can be imported as follows:

From the file menu, shown below, choose import, and then the Release/Skill pop-out.



You will see the following dialog.

A screenshot of a cell phone

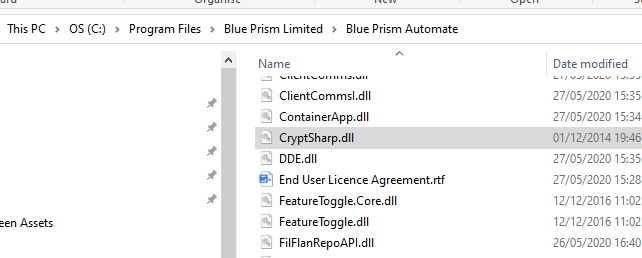
Description automatically generated

If you know the path to where you have the VBO stored, enter it into the text box, or navigate to it, using the browse button and file dialog that appears.

Once installed, your Blue Prism Studio should have the following entries in processes and objects.

|  |  |  |
| --- | --- | --- |
|  |  |  |

The CryptSharp.dll (available from the nuget link above) should be placed into the Blue Prism Automate working directory. This is shown below.



Once you have all this setup, you are good to go.

# Using the Asset

## Blowfish

The Blowfish action, uses the Blowfish algorithm to encrypt a supplied password or passphrase.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to encrypt using the blowfish algorithm. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **ciphertext** | The resulting ciphertext or hash output | Text |

## Blowfish Verify

The Blowfish Verify action takes in a password and a previously completed hash and they are compared. If they match, an isValid result of True will be returned.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to be verified. | Text |
| **Ciphertext** | The ciphertext to verify the password against. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **IsValid** | A true/false value stating whether the password hash matches that of the supplied ciphertext. | Flag |

## ExtendedDESEncrypt

The ExtendedDESEncrypt action, uses the ExtendedDES algorithm to encrypt a supplied password or passphrase.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to encrypt using the ExtendedDES algorithm. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **ciphertext** | The resulting ciphertext or hash output | Text |

## ExtendedDESEncrypt Verify

The Blowfish Verify action takes in a password and a previously completed hash and they are compared. If they match, an isValid result of True will be returned.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to be verified. | Text |
| **Ciphertext** | The ciphertext to verify the password against. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **IsValid** | A true/false value stating whether the password hash matches that of the supplied ciphertext. | Flag |

## LDAPEncrypt

The LDAPEncrypt action, uses the LDAP algorithm to encrypt a supplied password or passphrase.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to encrypt using the LDAP algorithm. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **ciphertext** | The resulting ciphertext or hash output | Text |

## LDAPEncrypt Verify

The LDAPEncrypt Verify action takes in a password and a previously completed hash and they are compared. If they match, an isValid result of True will be returned.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to be verified. | Text |
| **Ciphertext** | The ciphertext to verify the password against. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **IsValid** | A true/false value stating whether the password hash matches that of the supplied ciphertext. | Flag |

## MD5Encrypt

The MD5Encrypt action, uses the MD5 algorithm to encrypt a supplied password or passphrase.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to encrypt using the MD5 algorithm. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **ciphertext** | The resulting ciphertext or hash output | Text |

## MD5Encrypt Verify

The MD5Encrypt Verify action takes in a password and a previously completed hash and they are compared. If they match, an isValid result of True will be returned.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to be verified. | Text |
| **Ciphertext** | The ciphertext to verify the password against. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **IsValid** | A true/false value stating whether the password hash matches that of the supplied ciphertext. | Flag |

## PhpassEncrypt

The PhpassEncrypt action, uses the Phpass algorithm to encrypt a supplied password or passphrase.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to encrypt using the Phpass algorithm. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **ciphertext** | The resulting ciphertext or hash output | Text |

## PhpassEncrypt Verify

The PhpassEncrypt Verify action takes in a password and a previously completed hash and they are compared. If they match, an isValid result of True will be returned.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to be verified. | Text |
| **Ciphertext** | The ciphertext to verify the password against. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **IsValid** | A true/false value stating whether the password hash matches that of the supplied ciphertext. | Flag |

## SHA256Encrypt

The SHA256Encrypt action, uses the SHA256 algorithm to encrypt a supplied password or passphrase.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to encrypt using the SHA256 algorithm. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **ciphertext** | The resulting ciphertext or hash output | Text |

## SHA256Encrypt Verify

The SHA256Encrypt Verify action takes in a password and a previously completed hash and they are compared. If they match, an isValid result of True will be returned.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to be verified. | Text |
| **Ciphertext** | The ciphertext to verify the password against. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **IsValid** | A true/false value stating whether the password hash matches that of the supplied ciphertext. | Flag |

## SHA512Encrypt

The SHA512Encrypt action, uses the SHA512 algorithm to encrypt a supplied password or passphrase.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to encrypt using the SHA512 algorithm. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **ciphertext** | The resulting ciphertext or hash output | Text |

## SHA512Encrypt Verify

The SHA512Encrypt Verify action takes in a password and a previously completed hash and they are compared. If they match, an isValid result of True will be returned.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to be verified. | Text |
| **Ciphertext** | The ciphertext to verify the password against. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **IsValid** | A true/false value stating whether the password hash matches that of the supplied ciphertext. | Flag |

## DESEncrypt

The DESEncrypt action, uses the DES algorithm to encrypt a supplied password or passphrase.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to encrypt using the DES algorithm. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **ciphertext** | The resulting ciphertext or hash output | Text |

## DESEncrypt Verify

The DESEncrypt Verify action takes in a password and a previously completed hash and they are compared. If they match, an isValid result of True will be returned.

### Input:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **password** | The password to be verified. | Text |
| **Ciphertext** | The ciphertext to verify the password against. | Text |

### Output:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Data Type** |
| **IsValid** | A true/false value stating whether the password hash matches that of the supplied ciphertext. | Flag |

# Support

This asset is provided free-of-charge by Blue Prism. Blue Prism does not provide formal support of this asset. Please direct any questions you have, related to this asset, to the Digital Exchange Community page:

<https://community.blueprism.com/communities/community-home?CommunityKey=1e516cfe-4d1f-4de9-a9eb-58d15bf38c81>