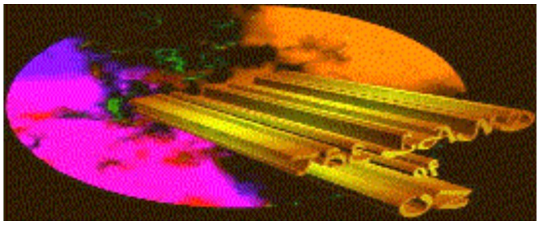
DAS Security For Windows
A Security Extension for Clarion For Windows



General Information

ISBN

Tin Man Software Corporation
P.O. Box 48823
Wichita, KS 67201-8823
(316) 264-3830
Homepage: http://www2.southwind.net/~tinman
@email: tinman@southwind.net
tinmanatoz@aol.com
74541.144@compuserve.com

Amierica Online: Tinmanatoz
Compuserve: 74541,144

Reichenberger Development Incorporated
Compuserve: 71324,1560

This is an updated version of a LIB we wrote for Clarion For Dos Version 3.
It has been updated for Clarion For Windows Version 1.5001 with new features added.

Introduction

DAS_Secuirty For Windows is a compilation of functions and templates which assist the developer in handling the difficult tasks of adding user friendly security to applications using the Clarion For Windows environment. The Library is made up of both 16 and 32 bit compiled LIB files as well as Utility, Extension and Procedure Templates.

We needed a better security system than the products currently available on the market. Our customers demanded more sophisticated security control. So we sat down and came up with this approach. Our customers loved the idea. One Security Administrator can easily maintain thousands of users, groups and resources incredibly fast. DAS Security is the most sophisticated securty product available for Clarion for Windows.

DAS_SECURITY was designed to allow you to implement a little security or a lot based on your needs. You may control access to the program only or to just about every thing else within the program. We selected to control access to the application features by the use of, what we call, "resources".

Resources are procedures, buttons, controls, entry fields, strings, etc, that you want to control access to. In a normal situation each procedure would be defined as a resource, and each control placed on the windows will have its own unique "resource" automatically assigned to it. Do'n let this scare you, we designed DAS_SecExport, an Application Template Utility, to create an export file automatically. This export file can then be imported directly into the Security Administration Program, which is included. This Security Administration Program is then distributed to your endusers so the assigned Security Administartor may assign access to your application.

Again, don't let this scare you, it is very simple to take an existing APP and add just the security you need. There is no handcodeing involved. Just point and click and you have the basic security already implemented. If you need to change the PROP:Attribute of any control in a window based upon the users security level, simply add the DAS_Security Procedure Template to that particular procedure and add the controls and thier respective PROP:Attribute to the Template (by point and click) and you are finished. Allow the on-site Security Administrator to handle the users, group and resource access. We even allow you to limit user or group access by the time of day.

The level to which you take this security system will vary with your needs. All the tools are included to make this the best security enhancement package on the market.

We think you will be very happy with DAS_Security For Windows. Our beta testers are rigorously testing all DAS_Security For Windows features across Windows 3.1, Windows 3.11, WFWG 3.11, Windows 95 and Windows NT platforms. We feel very confident in the stability of our product as we release it to the public. Tinman Software Corp gaurantees your satisfaction with our products or your money back.

As always, we encourage any, and all, comments regarding our products. Should you have any ideas for a new feature you would like added to DAS_Security For Windows, just let us know.

Adding DAS Security to the CW15 environment

REGISTER your templates in the CW15 environment.

1. Open CW15
2. Select Setup Menu
3. Select Template Registry

    (If you are updateing from an earlier version of DASTOOLS)

    - Scroll down and highlight CLASS DAS_ToolBox␡
    - Select the Unregister button
4. Select the Reigister Button
5. Go to the CW15\DASTOOLS directory
6. Double Click on the DASSECUR.TPL file.

Adding DAS Security to your Application

1. Open your application
2. From the Application Tree select Global
3. Select Extensions
4. Select Insert
5. Highlight DAS_Security
6. Click the Select Button
7. Check the Add DAS Security To Application Check box.

    Later on you can uncheck this box and no DAS Security Code will be added to the source code. This is helpful when trouble shooting compiler errors.

    When you re-check this box, all previous settings will be intact and the proper source code will be included.
8. Check if you need the Topspeed file driver added to your Project file.

    (DAS_Security requires the Topspeed File driver to be included in the Project, to prevent possible duplication errors at compile time, we have made it an option to have the templates automatically add this driver to your project.)

9. Click OK

Adding DAS Security to a Procedure

1. Highlight the procedure you would like to add the DAs security features to.
2. From the Application Tree select Properties
3. Select Extensions
4. Select Insert
5. Highlight 'Add DAS_Security Features to Procedures'
6. Click the Select Button
7. Click OK

This is all there is to it.

Applications

Resources are linked to applications and the security files may control more then one application. Users or Groups may be granted access to a resource for an application. In turn these Users can be assigned to Groups so even if the User does not have access to the resource but the Group does, the user will be allowed access. This simplifies the Security Administrators task of adding permanent or temporary users.

Protecting Procedures

Just by adding the "Add DAS_Security Features to Procedure" template to the Procedure you will have instant security provided for that procedure. If you desire Enhanced Security, simply check the "Enhanced Control/Prop" check box and go to the "Enhanced Security" tab and Insert the Control Used and its PROP:Attribute to change to if access is not allowed. Multiple controls can be defined as well as disabling the Insert, Change and Delete hot keys.

Protecting Browse Buttons (Manually)

For simplicity, on my account, we will just discuss the browse template. However, any and all DAS_Security Features can be easily added or handcoded (if neccesary) to any template available for Clarion for Windows.

Browses and other like procedures have resource for controlling access to insert, change, delete, view, etc. Access to these are controlled by having a resource defined for each of the browse functions.

1. Add the folowing source in the "Procedure Setup" source point.

```
IF DAS_CHECKACCESS( BROWSECUSTOMER ) THEN DO PROCEDURERETURN.
```

2. Add the following source in the "Preparing to Process the Window" source point.

```
IF DAS_CHECKACCESS( BROWSECUSTADD ) THEN ?COPY{PROP:Hide}=True.
IF DAS_CHECKACCESS( BROWSECUSTUPD ) THEN ?CHANGE{PROP:Disable}=True.
IF DAS_CHECKACCESS( BROWSECUSTDEL ) THEN ?DELETE{PROP:ReadOnly}=True.
IF DAS_CHECKACCESS( BROWSECUSTVIEW) THEN ?VIEW{PROP:Skip}=True.
```

3. Define the resource the resource in the security administrator program for the applcation.

Protecting Fields

{PROP:Hide} You may also want to protect fields. To protect a display field from being viewed.

```
   IF DAS_CHECKACCESS( FORMXXFIELD ) THEN ?FORMXXFILED{PROP:Hide}=True.
```

{PROP:Disable} To protect an entry field that is visible on the screen. then place the following code into the "Preparing to Process the Window" source point:

```
   IF DAS_CHECKACCESS( FORMXXFIELD ) THEN ?FORMXXFILED{PROP:Disable}=True.
```

{PROP:Skip} To skip over a control on a window add the following code to the "Preparing to Process the Window" source point:

```
   IF DAS_CHECKACCESS( FORMXXFIELD ) THEN ?FORMXXFILED{PROP:Skip}=True.
```

{PROP:ReadOnly} To prevent the contents of a control from being changed add the following code to the "Preparing to Process the Window" source point:

```
   IF DAS_CHECKACCESS( FORMXXFIELD ) THEN ?FORMXXFILED{PROP:ReadOnly}=True.
```

----- Security Administration Program -----

The Security Administration Program will be used by both you and the administrator at the users site. It is used to maintain the security system files, such as users, groups, resources, auditing, etc.

Security Files

To setup the security file the first time, run the security administration program as follows.

```
Security /SCREATE
```

This will prompt you for the key to be used to encrypt your security files. This will also setup a default user with the name of Master Operator password Master

There are 13 files in the security system as follows.

(These files are all combined and encrypted into one SECURITY.TPS.)

DAS_APP.TPS	- APPLICATION FILE NAME
DAS_PROC.TPS	- PROCEDURE FILE NAME
SEC_CTRL.TPS	- CONTROL FILE
*SEC_GPAA.TPS	- GROUPS TO APPLICATIONS
SEC_GRP.TPS	- GROUP FILE NAME
SEC_RESG.TPS	- GROUP RESOURCE ACCESS FILE NAME
SEC_PASS.TPS	- PASSWORD FILE NAME
SEC_RESN.TPS	- RESOURCE FILE NAME
SEC_RESU.TPS	- USER RESOURCE ACCESS FILE NAME
SEC_USER.TPS	- USER FILE NAME
* SEC_URAA.TPS	- USER TO APPLICATIONS
SEC_URGP.TPS	- USER TO GROUP FILE NAME
SEC_AUDT.TPS	- AUDIT FILE

System Control File

The following settings may be modified in the security system control file my using the Security Administration Program.

LOGINTYPE - Controls the type of login.

USENETID - Use the user netid as the login name.

PROMPTPSW - Prompt user for a password.

MAXTRIES - Number of retries for user name and password.

BEEPONERROR - Security system should beep on any security system error.

PASSSAVE - Number of user passwords to save.

PASSWARN - Password expire warning days.

DISABLELOGINS - Disable logins to all applications accept security.

ENVVAR - Name of environment variable for logins.

INVALIDUSER - Mark user invalid if login failed due to invalid password.

FAILNOTFOUND - Fail access if resource not found.

FAILSOFTMODE - Put system into fail soft mode (Turns off security).

LOGINONLY - Login in only not resource checking.

BACKUSER - Back door user id.

BACKPASS - Back door password.

AUDITALL - Audit all users.

AUDLOGIN - Audit logins.

AUDLOGOUT - Audit logouts.

AUDACCESSYES - Audit access allowed to resources.

AUDACCESSNO - Audit access denied to resources.

AUDPASSCHG - Audit password changes.

AUDSCNLOCK - Audit screen lock.

AUDSCNUNLOCK - Audit screen unlock.

AUDCONTROLUPD	- Audit control changes.
AUDAPPLID	- Audit applications access.
AUDUSERDEFINE	- Audit user defined entries.

Global Variables

The following variables are available at run time and may used to obtain information about the user and the security system. Refer to DASTOOLS.CLW for more information.

Variable Name	Description
DAS::SECOWNER	Key use to encrypt the securityfiles
DAS::SECFILEOPT	Security file error option
DAS::APPLID	ID of the current application
DAS::USERID	The users ID
DAS::NETID	The user network id
DAS::FIRSTNAME	First Name
DAS::LASTNAME	Last Name
DAS::INITIALS	Initials
DAS::TITLE	Title
DAS::LOCATION	Location
DAS::BPHONE	Work Number
DAS::BEXT	Phone extension
DAS::HPHONE	Home Phone
DAS::FPHONE	Fax Number
DAS::CPHONE	Cell Phone
DAS::PPHONE	Pager Number
DAS::PASSWORD	Password
DAS::PASSWORDPERM	Password Does Not Expire
DAS::PASSCHANGE	Allow Password Change
DAS::PASSINTER	Number Of Days Between Password change
DAS::PASS_EXPIRE	Password will expired on
DAS::PASS_CHANGED	Date Of Last Password Change
DAS::PASS_CHANGET	Time Of Last Password Change
DAS::USERTYPE	User Type
DAS::AUDIT	Audit Status
DAS::ADD_DATE	User Add Date
DAS::ADD_TIME	User Add Time
DAS::CHG_DATE	User Change Date
DAS::CHG_TIME	User Change Time
DAS::ACCESS_DATE	User Last Access Date
DAS::ACCESS_TIME	User Last Access Time
DAS::EXPIRE_DATE	Users ID Will Expire On
DAS::SECFPATH	Path To Security Files

----- PROCEDURES -----

Security Login - DAS_SECLOGIN

Logs the user on to the system. There are two types of logins standard and environment. Standard is a normal login were the user is prompted for a user name and password (optional). Environment uses a environment variable for the user name and will prompt the user for a password only (optional). The type of login is set in the security control file.

Prototype:

```
DAS_SECLOGIN(STRING)
```

Parameters:

```
RTNCODE   = DAS_SECLOGIN( APPLID )

APPLID    = The name of the application the user is logging into.

RTNCODE   = Return the following codes.
            0 - Login Allowed
            1 - Login Denied
```

Example:

```
CODE

IF DAS_SECLOGIN() THEN
  HALT(1)
.
```

Security Logout - DAS_SECLOGOUT

Logs the user out of the system and cause a audit record to be created if auditing is turned on. This is mainly used to force a audit entry to be created.

Prototype:

```
DAS_SECLOGOUT
```

Parameters:

```
NONE
```

Example:

```
CODE

DAS_SECLOGOUT
```

Security change password - DAS_SECPASSUPD

Allows the user to change their password. The last three passwords used are kept to force a new password to be used.

Prototype:

```
DAS_SECPASSUPD()
```

Parameters:

```
RTNCODE   = DAS_SECPASSUPD()

RTNCODE   = Return the following codes.
            0 - Password Changed
            1 - Password Changed Failed
```

Example:

```
CODE

IF DAS_SECPASSUPD() THEN
  !!!!!! Do Something
.
```

Security check access - DAS_CHECKACCESS

Checks user access for resource name being passed.

Prototype:

```
DAS_CHECKACCESS(STRING,<BYTE>)
```

Parameters:

```
RTNCODE   = DAS_CHECKACCESS(RESOURCE,MESSAGE)

RTNCODE   = Return the following codes.
            0 - Access allowed
            1 - Access Denied

RESOURCE  = The name of the resource as defined in the resource file.

MESSAGE   = Flag to display access denied message.
            0 - Do not display message (default if omitted)
            1 - Display message
```

Example:

```
CODE

IF DAS_CHECKACCESS( SECURITY )
  !!!!!! Do Something
.
```

Security lock screen - DAS_SECLOCKSCN

Locks the screen. The user must enter their password to unlock it.

Prototype:

```
DAS_SECLOCKSCN
```

Parameters:

```
NONE
```

Example:

```
  CODE

  DAS_SECLOCKSCN
```

Security create audit entry - DAS_SECAUDIT

Create a audit file entry.

Prototype:

```
  DAS_SECAUDIT(STRING,<STRING>)
```

Parameters:

```
  DAS_SECAUDIT(AUDITID,COMMENT)
```

AUDITID = A three byte id to identify the entry, the following are reserved by the security system.
(LIO) LOGIN OK
(LID) LOGIN DENIED
(TOD) LOGIN DENIED TIME OF DAY
(LGO) LOGOUT
(ACY) ACCESS OK
(ACN) ACCESS DENIED
(PAS) PASSWORD CHANGED
(LCK) LOCK SCREEN
(LCU) UNLOCK SCREEN
(CTL) CONTROL FILE CHANGED
(ADR) AUDIT FILE RESET

COMMENT = A comment up to thirty bytes long.

Example:

```
  CODE

    DAS_SECAUDIT( USR , ANY COMMENTS )
```

----- UTILITY TEMPLATES -----

To use any of the following Utility Templates, follow these steps:
(Remember...You must Register the DAS_Security Templates to the CW15 Environment)

DAS_SecExport......Export Default Security ResourceNames to Export File

(Export Application Information to [Application Name].SEC file)

1. Load your APP into the CW15 environment.
2. View the Application Tree.
3. Press 'Ctrl-U', This is the HOT KEY to the 'Select Utility' Menu
4. Double Click any of the Utility Templates under 'Class DAS_Security'
5. To view the files produced by these templates, simply load the file into the Clarion Editor or Notepad.

DAS_SecurityProcedure.....Add DAS_Security Features to Procedures

Just by adding the "Add DAS_Security Features to Procedure" template to the Procedure you will have instant security provided for that procedure. If you desire Enhanced Security, simply check the "Enhanced Control/Prop" check box and go to the "Enhanced Security" tab and Insert the Control Used and its PROP:Attribute to change to if access is not allowed. Multiple controls can be defined as well as disabling the Insert, Change and Delete hot keys.

If you would like the Enhanced Security Controls automatically created for you, just check the "Auto Build Control Properties" check box and specify the default Prop:Attributes you want to assign then go ahead and generate the source code. Every control placed in the Window will have security implemented automatically. Now if this is too much security checking, simply go back into the DAS_Security procedure extentsion template and edit the "Enhanced Security" features. All the automatically built Control/Props are displayed for your editing. These may be deleted, altered or added to.

Adding DAS Security to a Procedure

1. Highlight the procedure you would like to add the DAs security features to.
2. From the Application Tree select Properties
3. Select Extensions
4. Select Insert
5. Highlight 'Add DAS_Security Features to Procedures'
6. Click the Select Button
7. Check the Security Type you prefer
   - Procedure Security Only
     Implementes Security Checking for the Procedure Only. No further codeing is needed.
   - Enhanced Control/Property Security
     Allows Control properties to be changed at runtime based on the users security access.
   - Auto Build Control Properties
     Fastest Security - Automaticly generates security for all controls in the window with the default Prop:Attributes you select. These may be edited after source generation in the "Enhanced Security" features.
8. Click OK

DAS_GlobalSecurity.....Add DAS_Security Features to Applications

Resources are linked to applications and the security files may control more then one application. Users or Groups may be granted access to a resource for an application. In turn these Users can be assigned to Groups so even if the User does not have access to the resource but the Group does, the user will be allowed access. This simplifies the Security Administrators task of adding permanent or temporary users.

Adding DAS Security to your Application

1. Open your application
2. From the Application Tree select Global
3. Select Extensions
4. Select Insert
5. Highlight DAS_Security
6. Click the Select Button
7. Check the Add DAS Security To Application Check box.
   Later on you can uncheck this box and no DAS Security Code

will be added to the source code. This is helpful when trouble shooting compiler errors.

When you re-check this box, all previous settings will be intact and the proper source code will be included.

8. Click OK