

**Padlock**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> Padlock	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY		January 18, 2023
<i>SIGNATURE</i>		

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Padlock</b>	<b>1</b>
1.1	main . . . . .	1
1.2	whatsit . . . . .	2
1.3	howuse . . . . .	2
1.4	probs . . . . .	2
1.5	howwork . . . . .	3
1.6	features . . . . .	3
1.7	defaul . . . . .	4
1.8	version . . . . .	5
1.9	more . . . . .	5
1.10	credits . . . . .	5

---

# Chapter 1

## Padlock

### 1.1 main

Padlock is a small utility that I programmed in Blitz Basic 2. ↵  
I was  
trying to get the hang of the intuition commands, and decided to make a  
locking program like this because :

A: It would not take long to implement the basic code, and  
B: It was easy!

So

What is Padlock?

How do I use it?

What problems could I have with it?

So how does it work then, I mean really?

Any extra features?

What versions have there been?

Will there be any more then?

And

Do I really want to see the credits then?

I suppose I'd better write a disclaimer.

I will not be held responsible for any damage caused by this program (I  
won't) ( I won't). It wouldn't anyway (hopefully).

This program is FreeWare. Anybody who tries to sell it will get a visit  
from Da Boys (I have contacts) (I will know) (I will).

It can be distributed freely as long it is distributed completely. I.E.

Files should be :

Padlock

Padlock.info (I like my icon, but if you make a better one I  
suppose you can use that)

Padlock.guide (This doc)

Padlock.guide.info

If you don't get these files then send for Da Boys.

Iain King.

N.B. Sorry for being crap at writing docs.

## 1.2 whatsit

Padlock is a locking utility. If you are a 'secrets' person and you do not want anybody else to see your work then this is the solution! It allows YOU! - the user to lock up workbench at the touch of a button. Background tasks will still run, but no user access will be available. This is handy if you are on a network and want to go for a cup of tea without having to Logout

## 1.3 howuse

Padlock is designed with ease of use in mind. In its simplest form it is incredibly easy to use. The inspiration for Padlock was another locking program, but its interface had a horrible array of buttons (they were called LOCK, I and H. I found this user interface baroque). For this reason I have kept my interface simple - just two buttons.

The first time you use Padlock one will contain the word 'Password'. This is the input button. Hit this and you will be able to type in your password.

Now press Lock to Padlock your workbench. Where'd it go!?  
Don't worry, simply type your password and \*Bingo\* back it comes.

## 1.4 probs

Problems, surely there are no problems! But alas there are. ←  
Because of

the

way it works  
a program like Yak (A very cool

kommodity) can bypass the protection.

This will (hopefully) be fixed in a later version, but for now you should dissable programs that allow you to switch between windows without using the cycle gadget, or close all windows while you lock the screen.  
Sorry.

Other locking programs suffer from similar bugs. The 'inspiration' for Padlock had a harder to spot, but more serious problem. If you are running WBVerlauf (yet another cool commodity) then the program can be bypassed by holding down the right mouse button until the pointer stops moving.

Padlocks problem could lead to a similar breakdown, but the terminator would have to be accessable from an already opened window. - Padlock only allows access to already opened windows, the other locker allows full access.

---

## 1.5 howwork

So whats it all about then, and I mean really. I don't know but here's how Padlock works. When you hit the Lock button Padlock opens two new windows. The first is a completely blank window with absolutely no gadgets. This effectively clears the screen and disables workbench, while still allowing background programs to run. It then opens a second window that contains a plug and the text input box. You can then type in words. Nothing will happen until you type in the correct password\*. Once this happens Padlock will terminate (I like that word!) the two windows, returning you to workbench.

\* The password is case sensitive. This has ups and downs.

Downs:

Case sensitivity is a horrible, horrible thing (try using a BBC-B).

Ups:

You know the case you used - the other person does not. This makes it a lot, lot harder to guess your password, especially if you use YoYo case (I.E. lIkE tHiS)

I thought case sensitivety in this case was not that bad a handicap, and well worth the added security.

## 1.6 features

Extra features currently added are:

Runs from CLI. Parameters are <Password> or SET, HELP, ?  
Help and ? display a little help message.

SET is used for the  
Default  
password

Padlock also includes a  
Default password option.

Extended features are accessed via the menu's. There are ↔  
three menus,

and these are:

Project

Lock : As in pressing the Lock button  
Quit : Quits Padlock

Config

---

```
Snapshot          : Saves current configuration
Set default       : Saves current password as default
Restore           : Restores Padlock to saved config
```

#### Options

```
Show/Hide password : Toggles whether password is visible in
                    full display and while entering it when
                    locked
(De)Iconify        : Shrinks and enlarges Padlock window
```

Snapshot will save all user definable details about Padlock except the default password. 'Set default' will do this. Snapshot therefor saves:

```
X/Y coords of normal window
X/Y coords of small window
Whether password is visible or not
Whether window is small or large
```

These will be restored on startup. Alternatively, selecting the Restore menu option will restore Padlock to the previously saved options.

Show/Hide password is handy for those who like that little bit extra security.

(De)Iconify means you can get rid of Padlock (or at least make it as small as possible) if you have a crowded Workbench.

All prefs, including default password are saved as 'S:Padlock.config'. Please do not mess with this by hand. If Padlock gets screwy about the prefs (unlikely) then simply delete the file 'S:Padlock.config'. When next run Padlock will default to my in built prefs, and you can set it up and save it for yourself again.

N.B All menu functions can be accessed while in iconified mode.

## 1.7 default

The default password allows the user more speed in running Padlock. It means that rather than having to type your password in every time you run Padlock, you get a default for the password.

When Padlock is run from Workbench it automagically looks for your default password and uses it. If Padlock is run from the CLI and instead of a password you type 'SET' then it will look for your default password and use it.

Note: If you run from Workbench but there is no config file then Padlock defaults the password to 'Password'

Cool huh?!

---

## 1.8 version

There have not been many versions of Padlock (as if you hadn't guessed), but here you go :

Version 1.00	First release (to self)
Version 1.01	Some bug was fixed ( Input box I think )
Version 1.02	CLI startup added, as well as access to Padlock.Password
Version 1.03	Quit and Config menus added. All file access to S:Padlock.config
Version 1.04	Options menu added. Saves these extra features into config file
Version 1.05	Show/Hide Password now effects the entering of the password when in 'locked' mode

## 1.9 more

Will there be any more then. Yes! I would like to make the following improvements :

Some way to automatically disable Yak and others like it (Sorry)

I have attempted this, not by dissabling the commodities (not very nice), but by doing things with the windows and mouse, such as randomly moving the mouse pointer around the screen, and every few seconds bringing the two Padlock windows to the front. I haven't found a way of doing this that did not lead to it crashing. Sorry.

Head down Iain!

Any other ideas would be appreciated (but you can only take this idea so far)

## 1.10 credits

If you must.

Program, design (ha ha), and implementation by Iain King.

Written in Blitz Basic 2, which is by Acid software.

They also made the awesomely cool Skidmarks, and Guardian, and RoadKill, and Super Skidmarks, and.....

basically, their cool.

Blitz Basic 2 was given away free with Amiga Format a while ago, but to compile it I used a friends registered version. I should be getting my own soon (hopefully).

Iain King

---



Blairs Croft  
Creetown  
Newton Stewart  
DG8 7DF

<Now at Strathclyde University, but can be reached at above address>

EMail at : [iking@cs.strath.ac.uk](mailto:iking@cs.strath.ac.uk)

Bye Bye.

---