iAlertU

The alarm system for your Mac
© Randy Green
iAlertU v0.79
sourceforge.net/projects/ialertu

[Release Notes](#)

Welcome!
iAlertU is a utility that turns your MacBook into a motion sensitive remote controlled alarm system

System requirements:

MacBook or MacBook Pro, OS X 10.4 or higher, Quicktime 7.1.3 or higher, and a valid email account to use iAlertU's email functionality. iAlertU may now work on other Macs but has not been tested.

Installation
1. Drag the iAlertU icon to your applications directory.

2. Launch the iAlertU application and you with see the icon in your menu bar.

3. To arm the alarm system press and hold the menu button on your Apple remote control once and you should hear one chirp indicating the alarm is armed. You may also select the "Arm system" from the iAlertU menu to arm the system.

you must enable the "Enable arm from menu" checkbox in Preferences under the the General tab, and you MUST provide an arm/disarm password.

4. To deactivate the alarm press and hold the menu button on the Apple remote control once and you will hear 2 chirps and you will be returned to your desktop. To disarm the iAlertU without the remote you can enter your password using the keyboard.

Using the keyboard to arm and disarm iAlertU
iAlertU can be armed and disarmed without the use of an Apple remote control by selecting the "Enable arm from menu" checkbox in the General tab of the preferences window and entering a password.

Note that it is not possible to enable the "Arm" menu item without having provided an arm/disarm password.

With the above done, you may arm iAlertU by selecting "Arm System" from the iAlertU menu and entering the "arm" password.

To disarm from the keyboard, simply re-enter the "arm" password whenever you see the full-screen iAlertU logo. If you mistype, you will trigger the alarm, however you can simply begin entering the password again to disarm.

Note - To set up your MacBook to only respond to your remote control it must be paired. To do this aim your remote at the infrared port on the front of your MacBook. Next, press and hold the Play/Pause button and the MENU button at the same time until the paired graphic appears on your screen. To unpair your remote launch System Preference select the Security icon and press the Unpair button.

Configuring a Login Item

iAlertU into the Mac's login item list. This can also be used to effectively specify the -arm option (see below).

If you want to do this manually however, please follow the instructions below as they provide a couple of other features.

1. Start Automator, and create a new Application Workflow.
2. Add a "Run Shell Script" item to the workflow.
   Place the following line in the script:

   open -n /Applications/iAlertU.app --args <option>

   where <option> is one of:

   -check tells iAlertU to check if it was disarmed properly previously. If this option is used, and the application was not disarmed properly, then it will attempt to take another photo, and then send any photo's taken since the last arm event.

   -arm tells iAlertU to automatically arm immediately. This can be overriden from the preferences window by un-ticking "Allow automatic arming" in the General tab.

   -snap tells iAlertU to automatically take a photo and exit (unless -arm has been specified). This option requires the -pass option to be provided

   -pass <password> provides iAlertU with the unarm password. iAlertU requires this for some command line options where it is important to insist that the person starting iAlertU is authenticated.

3. Save the workflow to somewhere like Documents
4. Add a login item to your user preferences using the newly saved workflow.

may not be supported with older versions of Mac OS.

Using the -snap command line option
Version 0.6 added a new command line option that allows you to, from a command line, tell iAlertU to start, take a photo and exit.

This is done by providing both the -snap and -pass command line options. The -pass option is required because you don't want just anyone connecting to the Mac and requesting a photo as this could lead to misuse and privacy issues.

So, so take a snap from the command line use:

```
    open -n /Applications/iAlertU.app --args -snap
-pass <password>
```

where <password> is your unarm password.

Configuring Email to use Gmail accounts
A couple of people have asked for a little guidance on how to configure iAlertU to use their Gmail accounts. Below is a screen-shot of the Email Preferences panel. Just replace "yourgmailus-ername" with your Gmail username.

| Alarm | Camera | General | **Email** | Advanced |

☑ Email alarms

Email to: [yourgmailusername@gmail.com]

Email from: [yourgmailusername@gmail.com]

Mail server: [smtp.gmail.com]

☑ Use SSL Authentication   SMTP Port: [465] ↕

Mail server username: [yourgmailusername]

Mail server password: [••••••••••]

☐ Allow blank mail server password

☑ Email image(s) (requires camera to be enabled)

[Send Test Alert Message]

v 0.69

For Hotmail, the settings are not exactly obvious. The screen-shot below should help:

| Alarm | Camera | General | Email | Advanced |

☑ Email alarms

Email to: yourhotmailuser@hotmail.com

Email from: yourhotmailuser@hotmail.com

Mail server: smtp.live.com

☐ Use SSL Authentication   SMTP Port: 587

Mail server username: yourhotmailuserid@hotmail.com

Mail server password: •••••••

☐ Allow blank mail server password

☑ Email image(s) (requires camera to be enabled)

(Send Test Alert Message)

v 0.69b3

Script Support

As of version 0.69, iAlertU supports the ability to execute an Applescript file when any of the following events occur:

2. iAlertU is disarmed
3. An alarm is triggered
4. An alarm stops

This support was provided in response to suggestions and some generously supplied sample code from the iAlertU user community.

Things to note are that iAlertU will not allow you to specify an Applescript that is not already compiled, or that doesn't handle at least one of the above events.

Allowing the execution of an external script like this opens up a lot of potential opportunities for people to enhance iAlertU themselves. It also opens up the potential security holes. In adding this feature, we have endeavoured to make it as safe as possible.

Once a script has been specified within iAlertU, a copy of it is made, and encrypted in an attempt to prevent other people changing what the script does. This means that if you are busy debugging a script for use with iAlertU, be sure to tell iAlertU to reload the script each time you make a change.

One potential problem with the use of scripts is that if, having setup a script, and the encrypted copy of the script is deleted or corrupted somehow, iAlertU will not try to execute it. This means that even though you can get iAlertU to do other neat things via the scripts, it is recommended that you also continue to get iAlertU to send out emails and take photo's as it has always done. This way, if the script is disabled somehow, iAlertU will continue to sound the alarm and send out emails.

With an Applescript in place, it gives those of you who are able to write such scripts, the ability to get iAlertU to take additional actions when any of these events occur. One user does this to get Skype to call him when an alarm triggers. Another user has suggested using this to send an SMS to your mobile/cell.

separate download from the iAlertU source forge website. Please see within that file for a little more detail on what events are available. This may be added to over time. It is not my intention to provide detailed functionality within these scripts; that's for you.

| Alarm | Camera | General | Email | **Advanced** |

☐ Run script for events:

Script path:

~/Documents/Programming/scripts/ialertutrig   ▲

(Reload)

Device Filters

Ignore device state changes for the following device names:

Device name
IOHIDLibUserClient
IOUSB*
AirPort*
IOResource*
AppleSMCClient

[+][−]   When collecting, wait: [30 ▼] seconds.

(Collect potential triggering devices)

v 0.70

Device Filtering
A number of people have experienced intermittent problems
where iAlertU would trigger an alarm without anything "happen-

reason to trigger the alarm. In one instance the devices changing state were internal USB devices associated with the iSight camera.

In order to allow people to reduce this problem, a device filter table can now be configured within the Advanced tab. This filter table has existed for some time, however it is now user configurable.

In addition to being able to add your own filters, you can also ask iAlertU to attempt to collect any device names that for devices that change state whilst iAlertU is armed. Simply tell it how long to wait, and then click on "collect potential triggering devices", and iAlertU will arm for that period of time, during which it will record the name of any devices that change state.

Note that you can also place a "*" suffix on a device name to act as a wildcard.

 uAlertMe, and iPhone integration

With v0.71, iAlertU has a small embedded server that will allow a remote application such as uAlertMe for the iPhone/iPod touch to connect. Once connected to iAlertU, the client app can then 'see' what iAlertU knows about itself. The interface has been specifically designed such that it can be used *only* for iAlertU.

It does not give the client access to any other part of the Mac. It is also password protected, so it is as secure as your iAlertU password.

uAlertMe will work on the iPad, however it hasn't been customized to take advantage of it's larger screen yet. That will happen if the iPhone version is successful enough.

Documentation for uAlertMe, and how to configure it can be found at: http://www.pkclsoft.com/wp/?app=ualertme-for-ialertu

In order to allow uAlertMe to connect to your Mac, you need to be running iAlertU v0.71 or later.
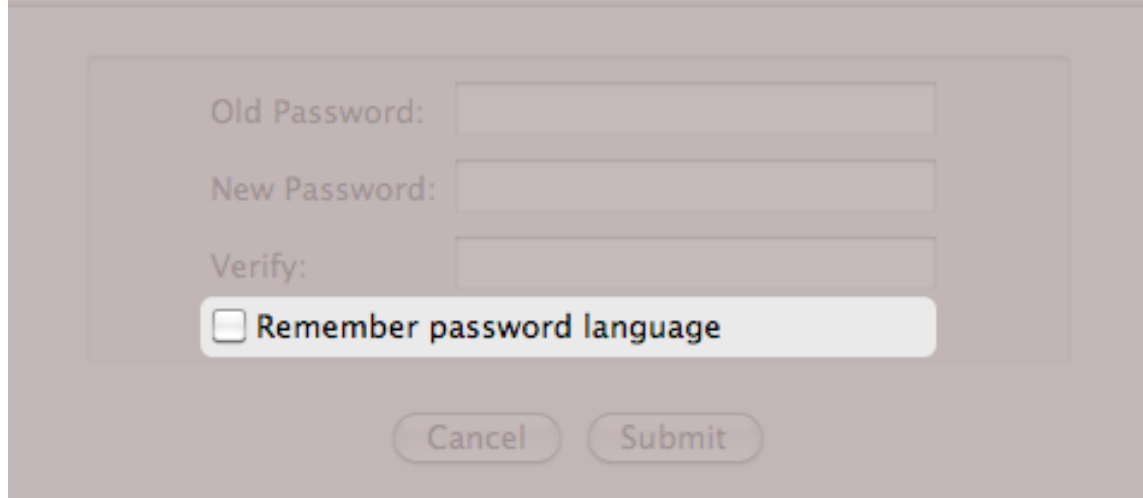
Once installed and running, you need to configure a couple of things within iAlertU:

1. Arm/Disarm password

To help you protect your Mac, iAlertU will not allow a connection from uAlertMe unless you know the correct arm/disarm password. As such, you need to configure this password within iAlertU on the General tab of the preferences panel (see below).
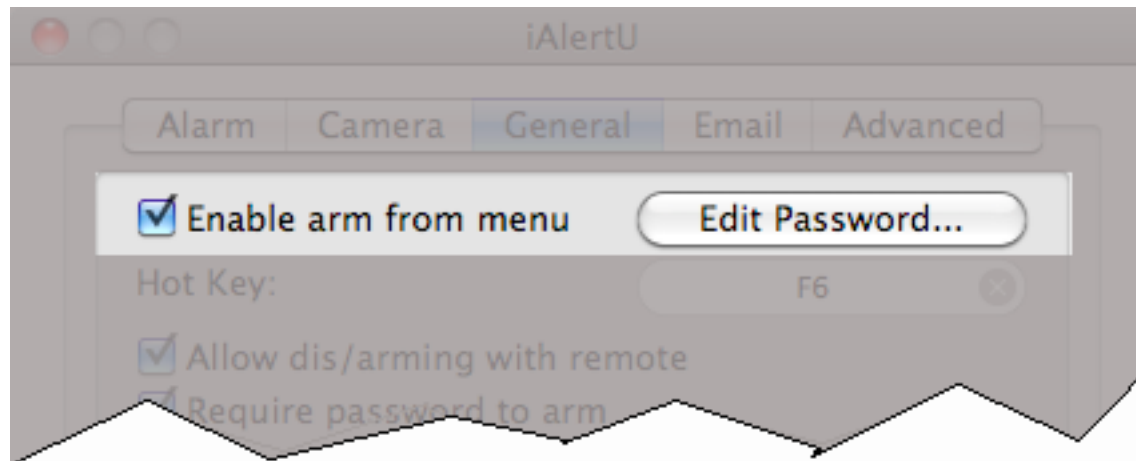
Clicking on "Edit Password" will allow you to specify the password. Be careful in this situation *not* to tick the checkbox in the password dialog.

Old Password:

New Password:

Verify:

☐ Remember password language

Cancel   Submit

2. Configure the uAlertMe interface

uAlertMe requires a way to connect to your Mac. For this, you need to configure iAlertU to run a small server that will listen for connections from uAlertMe. This server uses can use Apple's Bonjour service to advertise itself, or if you prefer, it can use a simple TCP/IP socket. If you are wanting to connect to iAlertU over the internet, then you will need to have your Mac configured with a static IP address, or have it configured to use Dynamic DNS. Bonjour works very easily on a local network, however getting it to work over the wide-area network (the internet) can be problematic.

Within the iAlertU advanced tab, there are several settings or preferences that you need to configure. If you have already read about configuring uAlertMe, you'll recognize that these mirror the connection settings there.
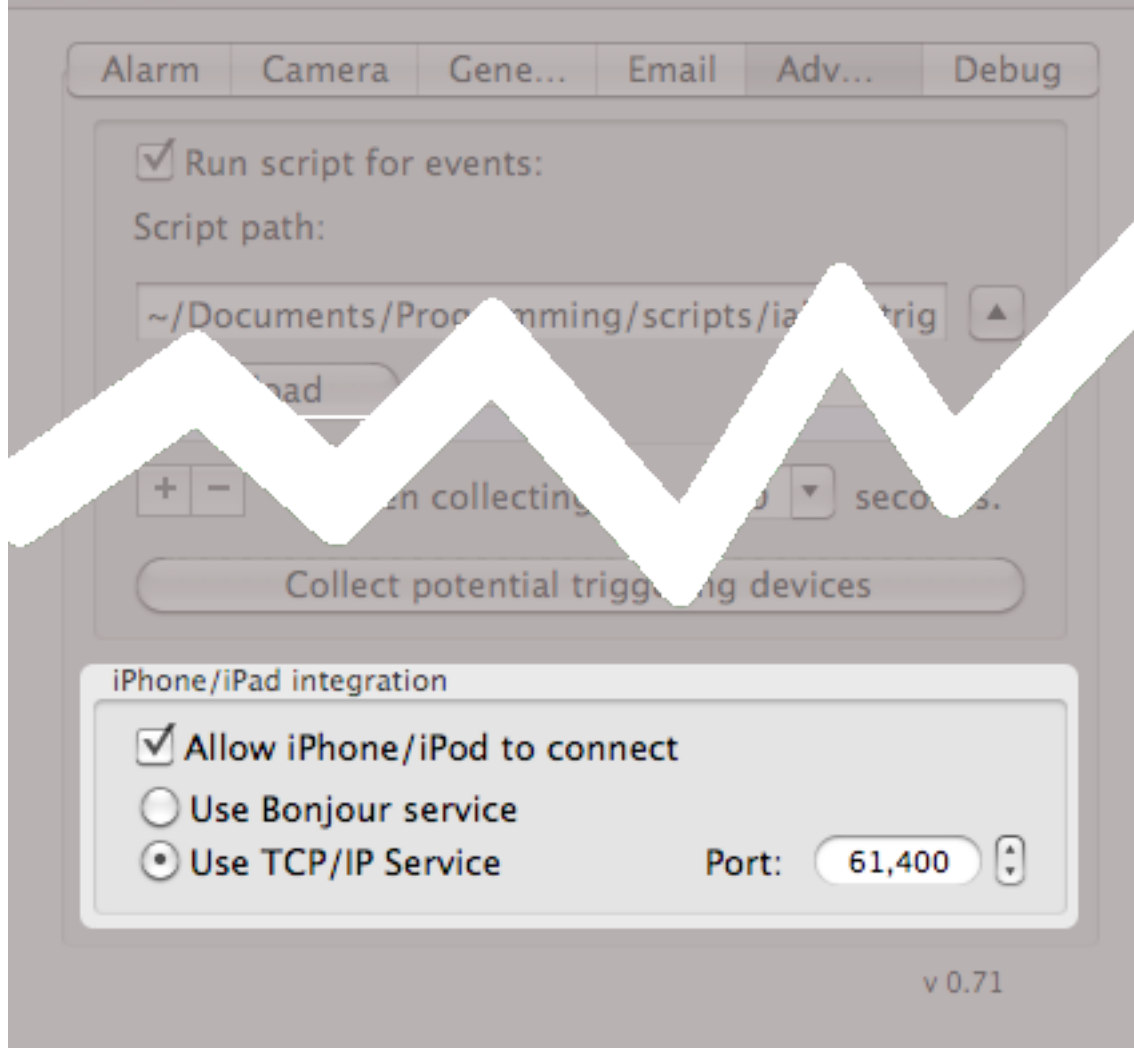
Ticking "Allow iPhone/iPod to connect" will turn the server on, using whatever settings are below. Any changes to the settings will cause the server the disconnect from uAlertMe, requiring a new connection to be established. If iAlertU is armed, then the server cannot be turned off or altered; this is to ensure that a thief can't stop you from accessing your Mac via uAlertMe.
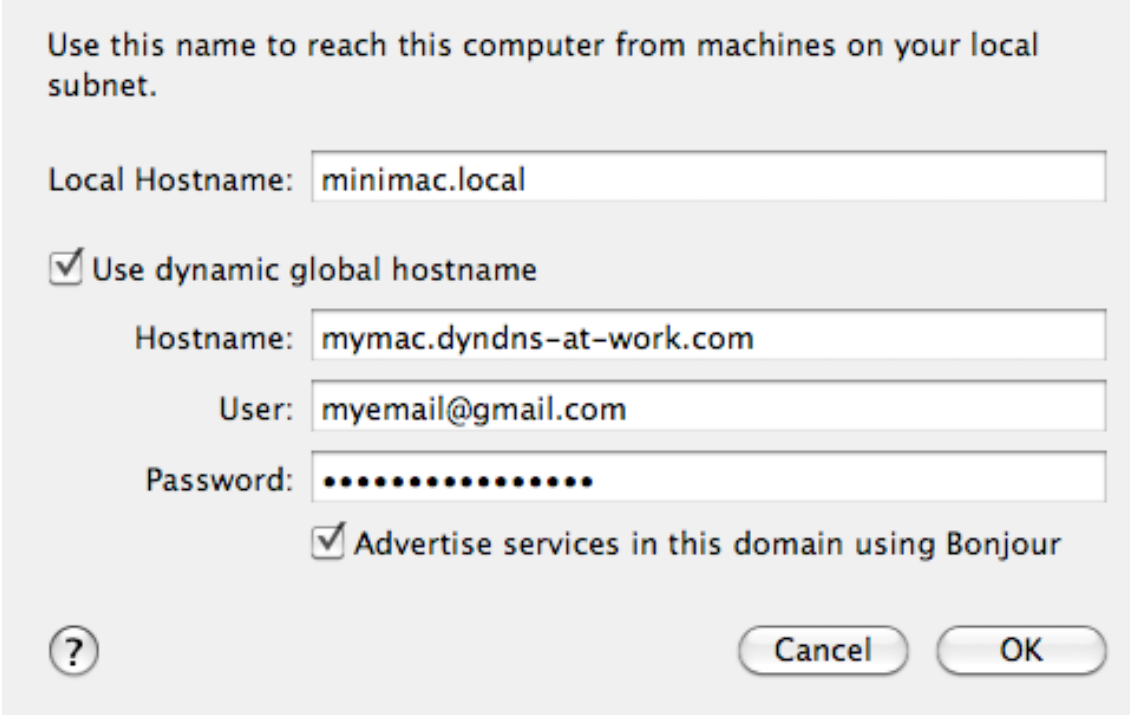
Ensure that, if you are using TCP/IP, the port number matches exactly, the port number you're using in uAlertMe.

3. Configure your Mac's sharing preferences

iAlertU to run it's server, and you need to tell the Mac how to advertise itself on the network.

For this, you will need to open the System Preferences application your Mac, and select "Sharing". With this selected, click on "Edit" to edit your Mac's computer name.

This will open a dialog such as:

Use this name to reach this computer from machines on your local subnet.

Local Hostname: | minimac.local |

☑ Use dynamic global hostname

Hostname: | mymac.dyndns-at-work.com |

User: | myemail@gmail.com |

Password: | •••••••••••••••• |

☑ Advertise services in this domain using Bonjour

(?)   (Cancel)   (OK)

In this dialog, the "Local Hostname" contains the value you would use if you want to use uAlertMe to connect over a local network.

If you want uAlertMe to connect over the internet, then you will either need a static IP address for your Mac, or you will need to set up a dynamic address via a website such as http://www.dyndns.com/.

If you are using dynamic DNS, then tick "Use dynamic global hostname" and configure the "Hostname", "User" and "Password" fields as per your dynamic DNS account.   If you want to use

ing wide-area Bonjour can get quite complicated, and I refer you to http://www.dyndns.com/ for more information on how to set it up.

Known issues/limitations:
A number of people have noticed that with a plug in the head-phone socket of the Mac, the alarm sound is not played on the speakers. This is a problem that differs from one MacBook(Pro) version to another. At this time, without some form of low-level hack I don't believe it's possible to bypass the headphone socket on a MacBook. It may be possible on some MacBookPro's, but without access to such a machine, I have no way to get this working. A future version will attempt to force the alarm sound out via the speakers if they can be found, in case the Mac is cur-rently using bluetooth headphones, but again this requires hav-ing hardware to test with.

Uninstall
1. Delete iAlertU file
2. Delete ~/Library/Preferences/com.slappingturtle.iAlertU.plist
3. Delete ~/Library/Application Support/iAlertU
4. Delete ~/iAlertU_*.jpg (if any exist)
5. Remove iAlertU from the login items if needed.

Release Notes

New/Fixed 0.79
- New alarm sounds have been contributed and the existing sound has been cleaned up. Thanks to Stephen Angelini for these contributions.

New/Fixed 0.78
- Fixed another issue whereby it was possible to arm without having set a password by closing the lid or putting the Macbook to sleep.
- Fixed a number of issues related to the remote interface.

Lion.

- Mountain Lion: Added several screen related devices to the default set of devices ignored so that when the Macbook is put to sleep, screen devices being shutdown do not trigger an alarm.
- Fixed a number of device state monitoring issues that might have resulted in unnecessary alarms, or missed alarms.

New/Fixed 0.77

- Prevent arming without a valid disarm password.
- Optimise the interface to uAlertMe so that connections are quicker.
- Fixed an intermittent problem that could prevent photos being included in emails or displayed on the Camera tab of the preferences panel.

New/Fixed 0.76

- Added Norwegian translations, kindly contributed by Torkel Adrien Thibault Genet

New/Fixed 0.75

- Bug fix to prevent auto arming if the disarm password has not been set.
- Added support for Korean (thanks to Jeong)

New/Fixed 0.74

- New feature: Push Notifications have been added so that people that have uAlertMe installed on their iPhone/iPad/iPod Touch can received a push notification whenever the alarm is triggered on the Macbook. With v1.1 of uAlertMe, all you need to do is connect to v0.74 of iAlertU (whilst the Macbook has an internet connection), and then after that alarms will be sent to the iPhone.
- Bug fix: A problem with disarming after the macbook has been asleep has been fixed when the Macbook is configured to ask for a password after a sleep.
- Bug fix: If a second or more display is connected when the

to better facilitate new connections.

- Bug fix: Device state changes were not triggering alarms every time.
- Minor memory leaks plugged.
- New feature: Animated GIF files may now be used for lock screens.
- Improved handling of keystrokes when armed so that spurious alarms aren't triggered when typing in your password.
- For people wanting to try and get the latest code working on Leopard, I've reduced the dependencies on Snow Leopard functionality.
- Bug fix: Message unpacking from uAlertMe could cause a crash, though very rarely.
- Removed support for legacy (pre 0.70) password mechanism.
- Added Macbook serial number and IP address to information sent to uAlertMe.

New/Fixed 0.73

- Corrected an issue whereby it was possible for a fresh installation of iAlertU to enable arming from the menu without having set an arm/disarm password. Now, if no password has been set, and you attempt to enable the arm menu item, you will be required to enter a password. This should prevent people from arming iAlertU without some way to disarm it.
- Bug fix that corrected the encoding of emails which could cause problems on some installations if the system encoding was not US ASCII compatible.

New/Fixed 0.72

- Bug fix for those people that enable "Remember password language" in the password dialog. This bug would cause iAlertU to crash when you attempt to arm.
- Bug fix relating to iAlertU not behaving correctly after the Mac has been 'asleep'.

New/Fixed 0.71

est OS X releases.

- Bug fix to allow selection of a script file on a clean installation of iAlertU.
- New feature; added a server that can listen on a TCP/IP socket, allowing an incoming connection from a remote client, allowing that client to manipulate iAlertU, and be alerted when an alarm is triggered. See **uAlertMe** for more information.
- A number of potential memory leaks have been addressed. None of these would have been high frequency leaks, so it's not expected that they would have cause any serious problems.
- The improvements to the French and Spanish localizations that were supposed to be in v0.70 have now actually been done.

New/Fixed 0.70

- Bug fix to ensure that the motion sensor monitoring code is turned off when disabled.
- Bug fix to correct the selected snapshot in the drop-down on the camera tab.
- A number of corrections to the French translations were kindly contributed by Pierre Morin.
- A number of corrections to the Spanish translations were kindly contributed by Jan Smite.
- A new feature, allowing iAlertU to use Speech Synthesis to "say" something when the alarm goes off (thanks James Wheatley).
- A new option to prevent the alarm sound from stopping if the Mac is still being moved (thanks James Wheatley).
- Changed the way that iAlertU uses CoreLocation so that it uses less network bandwidth. Several users reported high network bandwidth usage by iAlertU, and this was because CoreLocation was trying to locate the mac too accurately. This release should fix this by turning off CoreLocation unless it's needed.
- Dutch translations have been completed, thanks to Edgar de Graaff.

"disable idle sleep when armed" feature is enabled.

- Added new option to allow the user to specify that the screen should flash when the alarm is triggered, regardless of whether the logo is used.
- Added new option allowing the user to specify a different logo image to be displayed when iAlertU is armed.
- Added new options allowing the user to specify different sound files to be played for arming, disarming, and when the alarm goes off.
- New feature/bug fix whereby some users experience intermittent alarms are now able to configure/train iAlertU to ignore specific devices.
- The mechanism used to encrypt the arm and email passwords has been changed. The reasons for is that I want to remove the dependency on SSCrypto which is not available on the iPhone/iPad platform. v0.70 will automatically update your existing passwords by decrypting with the old mechanism, and then re-encrypting them with the new. It *should* be an invisible change from the users perspective. If people have problems with passwords after updating, I recommend that they re-enter them manually to update the configuration manually.

New/Fixed 0.69

- Added new scripting feature (see Script Support above).
- Bug fix to allow proper disabling of motion sensor.
- Bug fix to prevent crash if you never set a mail password and then disable email.
- Bug fix to ensure that you can unlock the Mac after closing it's lid.
- Added a new option to disable the taking of screenshots.
- Add feature to remember the input locale of the disarm password.   A user who frequently changes from one keyboard input locale to another requested this. It works by remembering the locale used to enter the arm password, and then whenever you later have to enter the password (e.g. whilst arming, or when armed), changing the input locale to the same one. It will also revert the input locale

immediately prior to password entry.
- Added feature that prevents the user changing the arm password without knowing the current one.

New/Fixed 0.68
- Corrected a number of issues preventing the latest version of iAlertU from working on Mac OS Leopard (10.5.8). Most of these showed as problems with email.
- Corrected a bug in the generation of latitude/longitude values in the southern or western hemispheres where the degrees component was zero.
- Disable the various controls on the "Camera" tab when the camera is not in use.
- Corrected the formatting of the new HTML email in some cases when all information is not available.
- Corrected issues relating to sending email when the sending of images is disabled.
- Don't try to override the default Java JVM for email. On some platforms (10.5.8) this caused problems.

New/Fixed 0.67
- Corrected the handling of some unarm passwords as they could be impossible to enter without triggering an alarm.

New/Fixed 0.66
- Corrected the application build setting so that it is no longer necessary for the CoreLocation framework to be present. This means that iAlertU should now work on OS/X Leopard 10.5.8.
- Corrected the handling of password entry when armed and no logo is in use.

New/Fixed 0.65
- Corrected (finally) the problem with validation of passwords which was causing most of the crashes on startup. This problem also caused odd behaviour with the email tab of the preferences window and the "Arm" menu item.

appear after disarming.

- Allow iAlertU to continue to run on Mac's running 10.5.x which does not support the CoreLocation framework. On these Mac's, the location information in the email will not be filled in.

New/Fixed 0.64

- Corrected the layout of the emails so that when no geographical location is available, the table is completed properly.
- Corrected another error in the handling of uninitialized preferences.

New/Fixed 0.63

- Corrected a bug that prevented new installations of iAlertU from starting. Upgraded installations should still have worked.
- Corrected a bug whereby using the keyboard to disarm using a disarm password would trigger an alarm in error.

New/Fixed 0.62

- Corrected a spelling error in the Dutch translation.
- Corrected an issue with connecting to mail servers with non-default port numbers.
- Corrected an issue with connecting to some mail servers such as hotmail.The ability to blank multiple screens has been fixed. I thought this had been done in 0.6, but without the required hardware I had no way of knowing.
- The logo now fades to black as it used to and is displayed on each screen. In addition to this, the logo will reappear as soon as a key is pressed, or the mouse is moved.
- A new preference has been added to allow some people to disable the fading of the logo.
- iAlertU now interfaces with CoreLocation in an attempt to determine the geographic location of the mac when an alarm goes off. This information is included in the email, and is embedded in the images (both in the filenames, and internally in their metadata).

sounds has been split into two preferences so that the two sounds can be disabled independently.

- iAlertU can now be installed as a login item from the preference window on the General tab. You can also optionally instruct iAlertU to automatically arm when it starts as a login item.
- The layout of the General tab of the preference panel has been re-done in an attempt to group the preferences.
- On the Alarm tab of the preferences window, if you move the Mac in a way that would normally trigger an alarm, an icon will flash just below the sensitivity slider, and a sound will be played. The purpose of this is to provide a means by which you can see how much movement a sensitivity setting will allow before triggering an alarm.
- An old bug whereby the password dialog was not by default given keyboard focus has been fixed.
- A new option has been added allowing you to cause iAlertU to automatically arm when the lid is closed. This is great if you are working on the mac when someone steals it. As soon as they close the lid, iAlertU will arm, and trigger when they reopen it.
- A new option has been added allowing you to disable Idle sleep on the Mac. This means that whilst the display might sleep (and it will as of this version), the computer itself will remain awake unless you force it to sleep by closing the lid or using the Sleep menu item in the Apple menu. Setting this option on also has the effect of delaying the forced sleep by up to 30 seconds. The great thing about this is that if the thief closes the lid, and picks the Mac up, the sound should still play, even with the lid closed as long as the Mac is still awake. There have been a lot of people that have asked for iAlertU to disable forced sleep altogether. At this point in time, I don't intend to do this as there is arguably a risk of overheating.
- The entire sleep management mechanism has been reworked so that the above two new options can work together. In doing so, the screen is no longer "captured" in the same

play(s) from ever being allowed to sleep.

- An old bug whereby an attempt to take a photo after opening the lid caused iAlertU to freeze (which in turn meant that there was no way to wake up the computer without a re-boot) has been fixed.
- The Email tab on the preferences panel will now validate the various email settings. Those fields that have invalid values will be highlighted in red. If any fields are invalid, the test email button will be disabled, and iAlertU will be unable to send any emails out. When iAlertU is first started, if email is enabled, and any settings are invalid, then a warning dialog will be displayed requesting that you fix any errors.
- Emails are now sent out in HTML format, so they look a bit nicer, and the information is easier to read.
- All of the text in warnings, the email, etc that is currently English-only has been reworked to allow it to be translated easily in the future (if you have time to spare, please contact us).

New/Fixed 0.61

- Fixed a problem with the default shortcut key.

New/Fixed 0.6

With this release we have been able to merge in some new features and bug fixes provided generously by David Reitter.

- Fixed the bug where multiple displays were not being blanked when iAlertU is armed.
- Added a new hot key feature that allows iAlertU to be armed by simply pressing a hotkey that you define in the General preferences.
- Added a new preference that allows you to arm using a hot-key or the menu without requiring the entry of a password.

   Note: A password will still be required to disarm.

the Apple remote to arm/disarm iAlertU.
- Added German translations, thanks to David Reitter.
- Added Dutch translations, thanks to Edgar de Graaff.
- New feature. A new command line argument is accepted that can be used to start iAlertU, take a photo, and email it. This new feature was requested so that people capable of establishing a remote connection to their Mac after it has been stolen can get iAlertU to send emails out without necessarily being armed.
- New feature. iAlertU now takes a snapshot of the screen and includes this in the email.

New/Fixed 0.57b

Note that this release is simply a bug fix release to help a few people that have had problems running on 10.5.8 (Leopard), or with French as the default language.

Users of 10.4.x (Tiger) "may" find that it works for them as well. Let us know if it does, as we have no Tiger platforms on which to test this.

- Fixed a problem with the storage of mail passwords on Mac's using French as the default language.
- Fixed a bug where the Send Test Email button would remain enabled for English users if "Email Alarms" is un-ticked.
- Attempt to make iAlertU run on 10.5.8 by allow it to run without the CoreWLAN framework.

New/Fixed 0.56b
- Fixed a crash at startup if the Mac was configured to use Spanish as the default language.
- Fixed a crash at startup if the Mac is running 10.5 or older.
- Ensure that test emails can't be sent if the email config is in-complete.
- Add ability to send emails without inclusion of photos.
- Added "New folder" button of our own to the path selection panel so that the new folder dialog is no longer hidden un-derneath the preferences panel.

used to disarm.

- The arm menu is no longer enabled if there is no arm password set (it must not be blank).
- Fixed a bug where the alarm sound would never stop if the arm/disarm sounds are disabled.

New/Fixed 0.55b

- Fixed a bug whereby using iAlertU as a standard user would not allow the sending of emails.
- Fixed a number of issues with regards to the use of the iSight camera. This results in iAlertU only turning the camera on for short periods of time when the photo is actually being taken. This in turn means that other applications such as Skype can have access to the camera the rest of the time. The downside to this is that if the camera is in use by Skype (or similar), then a triggering event for iAlertU will not include a photo because only one application can use the camera at a time.
- Fixed issues with regard to the "Check for updates" feature of iAlertU.
- Fixed the way that iAlertU detects motion so that it is not so sensitive.
- Fixed an issue that could result in a triggering event endlessly flashing or crashing entirely (requiring a hard-reboot).
- Added a new feature allowing the arm/disarm sounds to be disabled via the General tab of the preferences window.
- Added the trigger reason to the emails.

New/Fixed 0.54b

- Fixed an issue where iAlertU would trigger as a result of an AirPort event such as the local WLAN changing state. This is currently believed to be the reason behind unexpected trigger events.
- Fixed the bug where iAlertU required a restart after having triggered once.
- New preference added to allow the user to disable the arm and disarm sounds.

on the command line. This new argument, when used with a login item, will force iAlertU to automatically arm.

New/Fixed 0.53b
- A check is now done to ensure that the stored unarm password is encrypted. If it isn't then the user is forced to reset it.
- Closing the lid no longer disables keyboard input for iAlertU which has been causing problems when the lid is opened. The act of closing the lid causes OS/x to enter sleep, so when iAlertU detects that sleep has ended, it now triggers the alarm. This way if the MacBook is stolen and the lid is closed, when the lid is reopened later on, iAlertU will trigger then, when there is a better chance of photographing the thief.
- Added the ability to specify where iAlertU will store it's snapshots. This defaults to the users home directory.
- Opening the preferences dialog no longer turns on the iSight camera.

New/Fixed 0.52b
- Passwords are now encrypted before being stored.
- iAlertU checks at startup that the correct Java version is installed and available (1.5).
- Added ability to keep multiple images and view them from Preferences window.
- Added ability to keep multiple images and view them from Preferences window. They can also be deleted from here.
- Added integrated email application that no longer depends on Apple Mail. Emails are now sent from within iAlertU.
- Added test email functionality.
- Added -check command line argument that will force iAlertU to check if iAlertU had previously been armed and shutdown without being disarmed. This, when coupled with an Automator login item allows iAlertU to send any images out the next time the Mac is logged into, effectively dealing with the possibility of someone using the power button to circumvent iAlertU.

tails to the emails that are sent out.

- Added auto detection of network availability if iAlertU has previously been started with -check argument when no network is available.
- Added with many thanks from other contributors, both French and Spanish UI (some dynamic alerts are still English only).

New/Fixed 0.30b
- Fixed iSight compatibility with Snow Leopard
- Added remote codes for Mac OS 10.6.2

New/Fixed 0.29b
- Simplified email screen
- Added trigger for device tampering
- Screen close, AC Adapter, and Motions sensors options are disabled for non portable machines.
- Added user option to enable and disable the alarm sounding at maximum volume.

New/Fixed 0.28b
- Fixed to work with both Tiger and Leopard

New/Fixed 0.27b
- Fixed iAlertU email bug where email was not being sent

New/Fixed 0.26b
- Fixed Apple remote bug in Leopard

New/Fixed 0.25b
- Open sourced under GPL license

New/Fixed 0.24b
- Photo email should be working better
- several other small bugs fixed

New/Fixed 0.23b
- Fixed bug in password window

- Improved the password entry functionality

New/Fixed 0.21b
- Password entry bug fixed
- Alarm sounds for 10 seconds after lid is closed because forced sleep can not be disrupted.

New/Fixed 0.20b
- Default preferences fixed
- Lock screen without logo now clear again

New/fixed 0.19b
- Captured photo can now be emailed automatically to a POP3 email account.
- Camera capture functionality improved.
- iAlertU should work on more Apple systems
- Users can now arm the system without the Apple remote.
- Added Sparkle framework for automatic updates.
- New icons by Kyle Kesterson

New/fixed 0.18b
- Fixed bug in code that was causing kernel panics.

New/fixed 0.17b
- When FrontRow is active iAlertU is ignored
- Power button, eject button and several other keys will now set off the alarm

New/fixed 0.16b
- Fixed F3 mute issue with MacBook machines

New/fixed 0.15b
- Coded optimization. CPU utilization has been significantly reduced in this release.

New/fixed 0.14b

New/fixed 0.13b

New/fixed 0.12b

- Wrong version was uploaded in .11b so .12b gets everyone in sync.

New/fixed 0.11b

- Added some debugging to log mute button activity for the MacBook owners
- iSight performance improved

New/fixed 0.10b

- F3 mute button disables correctly now when alarm is sounding
- Screen logo now fades to black after 10 seconds. This will soon be a user preference
- Alarm is now activated and deactivated by holding the menu button for approximately. By using the delayed menu button all the other applications(FrontRow, iTunes, iPhoto, etc) should work correctly sharing the remote with iAlertU.

New/fixed 0.09

- Alarm now activates when power adapter is removed

New/fixed 0.08

- Added basic camera capture functionality. The camera component still needs some work.
- Fixed some performance issues and this release should be more stable.

New/fixed 0.07:

- Should work better with FrontRow and again with iTunes.
- If dual monitors and using the logo the second monitor will black out

New/fixed 0.06:

- Preference panel now part of the overall menulet
- Made menu bar icon more Apple-esqe.
- Updated window texture to more modern look

- iAlertU now shares the remote with FrontRow without the need of a toggle
- Preferences added to turn on and off different sensors(motion, keyboard, and trackpad)
- Preference added to turn Lock Screen logo on and off
- Removed application timeout

New/fixed 0.04:
- UI worked on
- Code cleaned up, memory leak fixed
- Bug Report menu item put back in(Whoops!)

New/fixed 0.03:
- Created a main window with preferences
- Added user adjustable sensitivity slider
- Added user adjustable alarm duration
- Added checkbox to enable and disable remote control of iAlertU. This returns control of the remote back to the system while iAlertU is still running.
- Fixed popup windows so they get focus

New/fixed 0.02:
- Screen lock with logo when armed
- Alarm sounds on key press or mouse activity
- Volume returns to normal listening level after alarm is disarmed
- Alarm will now keep sounding for up to 30 seconds after lid is closed and continue after the lid is reopened
- Mute disabled when armed
- Control-tab disabled when armed
- New icons
- Cleaned up audio

New in 0.01:
- First beta release
- Forces system to un-mute when alarm is active
- Forces max volume when alarm is active
- Screen flashes when alarm has triggered

Purchasing
iAlertU is free but you are welcome to donate if you would like.
Just click the "Donate" on the iAlertU menu.

Disclaimer
iAlertU is provided as is. The Author is not responsible for any damage to the user's computer system or data and in no event will The Author be responsible to the user for any consequential, incidental, or indirect damages (including damages for loss of business profits, business interruption, lost of business information, and the like) arising out of the use or inability to use the The Software, even if The Author has been advised of the possibility of such damages. Because some states do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitations may not apply to you. Further, the consumer agrees that using this software in any way illegal, or even for the purposes of assessing its viability for illegal usage, is prohibited.

Credits

Sparkle Updater
Version 1.1
Copyright (c) 2006 Andy Matuschak

Thanks to Paul Ingendorf for fixing the mute issue with the MacBook non Pro machines.
www.timespace.org

Graphics and icons by
Colin Anawaty
http://www.cubedcompanies.com
and
Kyle Kesterson
http://www.iamintricate.com

motion.c

HIDRemote
Copyright (c) 2007-2009 IOSPIRIT GmbH
(http://www.iospirit.com/)
All rights reserved.
http://www.iospirit.com/developers/hidremote/

Special thanks to Pall Thayer creator of the Carpenter's Level
Dashboard Widget for all of his help getting the motion.c code
working on the MacBook Pro. His website is located at
http://pallit.lhi.is/palli/dashlevel/

Audio file enhancements.
Tom Zappala http://www.yeahtomdoesmyhair.com

BIG_alarm_1 & BIG_alarm_2©2012 Stephen Angelini. License is given for free
use and distribution of these sounds in the iAlertU project only. Any use of these
sounds in any other project, commercial or otherwise, is forbidden unless dis-
cussed with the author: contact mybizz_at_sympatico.ca

ShortcutRecorder
Copyright 2006-2007 Contributors. All rights reserved.
License: BSD

Thanks to http://caffeinatedcocoa.com/blog/ for their great post
on how to access meta data within image files
(http://caffeinatedcocoa.com/blog/?p=7 - comment-4952).

Thanks to **Rogue Amoeba LLC** for their assistance with the
audio code, and the integration of login item support.
http://www.rogueamoeba.com/

The CocoaAsyncSocket project for their wonderfully easy to use
sockets API.

The CocoaLumberjack project for their logging mechanism that
has been added to iAlertU as a part of integrating CocoaAsync-
Socket.

separately at:
http://www.pkclsoft.com/wp/?p=350

Translations
Spanish: Emmanuel Fernández Campillo, Jan Smite
Dutch: Edgar de Graaff
German: David Reitter
French: Pierre Rudloff, Pierre Morin
Korean: Jeong
Norwegian: Torkel Adrien Thibault Genet

Contributors:
      Peter <pitpalme@users.sourceforge.net>
      James Wheatley
      David Dauer
      Jesper
      Jamie Kirkpatrick
      Edgar de Graaff
      David Reitter
      Peter Easdown

Contact
      http://ialertu.sourceforge.net/

If you like iAlertU please consider donating to the project to help support the ongoing development.

http://sourceforge.net/projects/ialertu/

Thanks and enjoy.