Blast

Calculator Security Suite

User's Guide
And
Documentation

By Anthony Cagliano
http://clrhome.org/blastav

INSTALLATION & DEPENDENCIES

Welcome to the **Blast Calculator Security Suite**, an all-in-one collection of security-related tools for the Texas Instruments TI-84+ CE graphing calculator.

This suite requires the C libs by MateoConLechuga for file handling and graphics. It also needs the following memory availability:

~7500 bytes for program executable
22 bytes for settings file
4 bytes + 14 bytes per program for attributes file
Enough RAM to install current virus definitions
- All files created by this suite are kept in archive after initial creation.

To send this program to your calculator, you may use the Application TI Connect CE as well as a USB Type A to mini B cable. You will want to drag this program (BLASTCSS) over on top of the name of your calculator within TI Connect CE's GUI. You'll get a popup asking you what calculators to send to and whether to send to RAM or Archive. You should be able to just hit *SEND*.

Additionally, I recommend that you install the latest virus definitions file from the Blast CSS Project Page at http://clrhome.org/blastav. By clicking on the link near the top of the page, you'll be redirected to the place where you can download this file. You will need to send it to your calculator the same way as the main program.

USAGE & FEATURES

This program has quite a few nifty features that hopefully will afford for decent security against your friends pranking you and the possibility of networking becoming a thing on the CE.

Attributes Saving: First, this program records attributes about all programs and protected programs on your device. This includes the file name, size, and checksum (a cumulative sum of each byte of the program). You can generate this database by selecting the *Update Attributes File* option from the main menu of this program. The file generated is timestamped, and the suite will print the date the file was generated at the bottom of the screen, as well as an alert if the attributes file is becoming outdated.

Attributes Verification: Secondly, this program has the ability to check the attributes of the current state of files against their last recorded state. You can perform this check by selecting *Verify Attributes* from the main menu. Doing so will cause your calculator to look for every file in your database and compare the attributes to existing versions of those files. You will then be shown a list of every program or protected program scanned, by name. Next to that, you will see the words *size* and *checksum*. Those words will be printed in green if the attributes match and red otherwise.

Mal-code Scanning: Thirdly, the program has the ability to scan every program or protected program on your calculator, looking for byte sequences that can cause serious harm. If such a code is found, you are shown (1) The name of the program it is found in, (2) The offset

from the start of the program where the opcode is found, and (3) The address in memory the opcode is located at. These codes are saved in a specially-formatted *virus definitions file* that, when loaded by the calculator, tells it the byte sequence to look for and what that byte sequence does. (**Special thanks to Zeda for providing the first definition for this project**). The virus definitions for this project are community-sourced, meaning if you know of some byte sequence that does something bad, submit it! You can do so by going to http://clrhome.org/blastav/definitions.php. Like the attributes file, this file is also timestamped and an alert will be printed if you go too long without updating it.

System Time Restore: The last feature in this program is a silent one. I'm sure many of you have gone through the pain of setting up the date and time on your calculator, only to have it be erased when your calculator resets. Well this program changes that. Every time you see the main menu, the program has just saved your system time and date. If, during startup, this program notices a reset date, the date is silently restored to the last save. Simple as that.

Smart-Detect: Accessible from the *Advanced Settings* menu, this option installs a Program Run intercept hook. Before any program is run, a quick scan of the program against the virus definitions file occurs. If a match is found, the execution of the program is aborted. The hook is saved as an archived Application Variable, and every time BLASTCSS is run, the hook is repaired if necessary.

Other upcoming planned features are: (1) automated file attribute updating after editing program editor, (2)

firewall implementation, pending networking implementation on the CE.

TECHNICAL SPECIFICATIONS

This program utilizes the standard rootkit string detection/file integrity scan method used by Linux-style systems for malware detection. Whereas a positive on any one of these tests may not mean you are infected, a file failing both checks, or a positive in the virus scan and one or more files with invalid attributes may be evidence of an infected device and should be addressed. Further releases of this program will add the ability to quarantine (convert to appvar) any infected programs.

This program will also (in the future) utilize a state-wise firewall architecture to help protect users if the CE ever gains CALCnet or some other form of networking. This firewall will log any packet sent by your device and whitelist the recipient of that packet as an ESTABLISHED connection. Further inbound packets from that host will be allowed. After that, any inbound packet that matches a host on a REJECT list will be rejected (deleted before handling). Lastly, any packet that passes the REJECT test will have its headers verified. A packet with an invalid header will also be REJECTED.

For developers:
- To allow compatibility with the firewall of this program, your networking protocol must allow for hand-off to the firewall for processing of packets while packet information is available. Outbound connections will be logged to the state file as ESTABLISHED (state-wise connection). Inbound connections will be filtered, ESTABLISHED connections will be allowed, a series of rules will be

processed and a default policy will be applied. You will need to inform me of the configuration of your protocol's packets for calibration of the firewall.

Required Reserved Memory Areas
- MEMORY (Address Field 1) [24-bits]: Pointer to code to execute (outbound packet filter handler).

- MEMORY (Address Field 2) [24-bits]: Pointer to code to execute (inbound packet filter handler).

- MEMORY (Address Field 3) [24-bits]: Pointer to code to execute (firewall de-initialization).

Required Handlers within Code
- CODE: At termination of network set-up, check for *TIfw* application variable. If exists, hand off execution to start of appvar. Handoff should be a CALL. Code within the appvar will load the firewall's inbound and outbound filters and de-initialization code into the appropriate memory fields, and create the state-file.

- CODE: Within inbound packet routines to hand off control if Address Field 1 has a non-zero value. Handoff should be a CALL.

- CODE: Within outbound packet routines to hand off control if Address Field 2 has a non-zero value. Handoff should be a CALL

- CODE: Within network set-down, to hand off control to firewall deinitialization handler. Handoff should be a CALL. This zeroes all the Address fields and deletes the firewall state file.

** While the network is up, the firewall should now be active. Unfortunately, during protocol initialization, you need to handle loading the TIfw appvar yourself because I cannot guarantee its location will be the same. **

If you require more information, please do not hesitate to contact me at ac@clrhome.org.

LEGAL NOTICES

This program is, for obvious reasons, closed source. We at ClrHome Productions disclaim any and all liability for damage to your device that may follow from the intentional misuse of this program, including but not limited to: reverse engineering and modifying the main program, modifying a dependency, attempting to break the program.

As with any software of this kind, use of this software does not guarantee security but offers many layers of protection, including detection of dangerous code, alert of file modification, and more. This program is not a substitute for smart use of technology.

As an end user of this product, you are able to download and install it to your device, as well as share the programs or the ZIP, in whole or part, so long as a copy of this documentation is also provided and said files are unmodified. We cannot help you troubleshoot if you are using modified software.

CONTACT

Email: ac@clrhome.org
Twitter: @ACagliano
Facebook: https://facebook.com/acagliano
Website: http://clrhome.org

CHANGELOG

Version 0.9 beta
- Malware scan now shows, in addition to the name of the offending program, the location of the malware relative to the start of the program and the address at which it occurs.
- Files in the attributes database that are not found on the calculator during the attributes verification are still shown with the text "file not found" in orange.

Version 0.8 beta (unreleased)
- ProgRun hook appvar changed in name from *AVHook* to *AVPars*, for identification as the parser hook, due to planned implementation of an app change hook.
- Graphical glitches with menu returning fixed

Version 0.7 beta
- Alert for outdated database changed to yellow.
- Settings menu added, GUI update.
- Menu overflow and position save added.
- Firewall moved to Settings.
- Smart-Detect ProgRun interception added.

Version 0.6 beta
- Fix graphical issue where having a file in the attributes database not exist during verification caused results to display improperly.

Version 0.5 beta
- Fix bug in malware scan caused by improper use of ti_Open() rather than ti_OpenVar(), which caused the program to not actually scan anything.

Version 0.4 beta
- Malware scanner officially added to the suite of tools (formerly was a dud-menu option).

Version 0.3 beta
- *Checksums Database* renamed to *Attributes File* in menu.
- Ability to save and restore system clock added.
- Timestamps added to all databases, as well as ability to detect outdated databases.

Version 0.2 beta
- Bugfixes of file attributes saving and verification system. Skipping missing programs added.

Version 0.1 beta
- First official release. Ability to save checksum and size of all programs and protected programs, and verify them later, added.