

Overview

The Phoenix SLIC Tool is a new tool to insert a SLIC table into Phoenix BIOSes (original and Lenovo) using one of three main methods:

- 1) Original – currently implemented only for old/original Phoenix BIOSes. Uses PREPARE/CATENATE (or FP/FI) to generate the new BIOS. This is really for reference only, there have been reported bricks with this method.
- 2) SSV2 – main method recommended for most mods. Modules are inserted and changed without altering their offsets.
- 3) Dynamic – an existing SLIC table, or existing SLIC elements, are replaced.

It can also replace EXISTING SLIC tables in Insyde and Dell BIOSes.

Method

- 1) Flash, using the manufacturers recommended method, the unmodified BIOS of the same version. This allows the bootblock etc. to be updated.
- 2) Mod the BIOS using the tool.
- 3) Flash the SLIC'd BIOS from DOS. This ensures the bootblock is left intact and maximises the chance of a successful recovery if a problem was to occur (see below for more information).
- 4) You then still need to use the appropriate key and certificate for Vista and Windows 7.

Many thanks to Yen, Busykid508, 911medic, shakeyplace, str8, phrunt, rbjack and Zort for their invaluable advice and feedback. Special thanks to phrunt for experimental testing on hardware and to busykid508 for tirelessly using and giving feedback. Thanks also to middleton for guiding me through EFI modification.

**** This tool is freeware and may not be sold or redistributed for a charge ****

Notes:

- Please check for an existing mod first: <http://forums.mydigitallife.info/showthread.php?t=7500> and <http://forums.mydigitallife.info/showthread.php?t=5864>.
- BIOS flashing is not for the faint hearted. There is ALWAYS an element of risk. Please ensure you are familiar with the procedure, and have necessary floppy disks etc., to allow a recovery in case of a flash failure. **THIS IS ESPECIALLY TRUE FOR PHOENIX BIOSES.**
- We assume you have a legitimate edition of Vista or Windows 7. Piracy is NOT condoned.
- Please remove any softmods. For Vistaloader this includes repairing the boot sector and MBR of the hard disk.
- The tool requires the .NET Framework 2.0

Flashing – always best to flash from DOS:

- **For ACER Phoenix BIOSes – use the flash tool that comes with the BIOS**

- For Phoenix Bios: `ph161700 /X /FORCE /O /C /S <NameOfBIOSFile>`
(http://slics.myftp.org/Tools/Phlash_17/PH161700.EXE). Don't use the /BBL switch.
You can use winplash for x32 (ensure Advanced=1 and Hide=0 in [UI] section of phlash.ini).
Do NOT use winplash for x64.

- For Insyde – use the DOS flash tool that comes with the original BIOS or
`flashit <NameOfBIOSFile> /all`
(<http://slics.myftp.org/Tools/FlashIt/FlashIt.rar>)

- For Dell: use either the modified EXE created by the tool or run the original BIOS EXE with
'-readgzfile BIOS.GZ' (minus quotes) as command line

- For Dell BIOSes with Multiple GZIPs (some Optiplex models) – look in log – use the Dell Client Configuration Utility (DCCU).

Flash recovery information:

- Read here - <http://www.biosman.com/biosrecovery.html>.
- Also read here - <http://forums.mydigitallife.info/showthread.php?t=2105>
- For Insyde read here - <http://forums.mydigitallife.info/showthread.php?t=7033>
- For ACER read here - <http://forums.mydigitallife.info/showthread.php?t=13095>
- Ensure that the version on the disk is the (unmodified) version you are trying to flash with SLIC - avoids problems with bootblock flashing.
- **PLEASE ENSURE YOU HAVE CREATED A PHOENIX CRISIS DISC OR INSYDE RECOVERY DISC BEFORE ANY FLASHING**

SLIC files - The program needs:

- A 374 byte SLIC file. Checksum is automatically corrected.
- For 2.1 SLICs look in <http://forums.mydigitallife.info/showthread.php?t=5952>.

RW Everything reports:

- **When performing ANY Phoenix or Insyde mod YOU WILL NEED a RW Everything report** (see <http://forums.mydigitallife.info/announcement.php?f=33>) from the UNMODDED BIOS to load into the tool before modding the BIOS.
- For phoenix, when the RW report is loaded the tool will automatically select SSV2 or dynamic depending on the presence of a SLIC table in the report:
- If a SLIC table is present -> dynamic is selected
- If a partial SLIC table is present -> SSV2 is selected and 'Replace existing SLIC elements is also selected
- If no SLIC table is present -> SSV2 is selected and 'Replace existing SLIC elements is not selected

Vista Keys and Certificates:

- Keys can be found here - <http://forums.mydigitallife.info/showthread.php?t=2581>
- Certificates can be found here - <http://rapidshare.com/files/118119842/39-CERTS.zip>

Windows 7 Keys and Certificates:

- Keys can be found here - <http://forums.mydigitallife.info/showthread.php?t=10370>
- Certificates can be found with SLICs here - <http://forums.mydigitallife.info/showthread.php?t=5952>

General Options

Don't alter any ACPI tables

No tables, whether in the main ACPI (BIOSCODxx) module or in separate ACPIxx modules, will be altered.

Note – existing SLIC tables/elements will still be changed.

Don't alter any ACPI tables or OEM/Table ID strings if SLIC IDs match

If the OEM/Table ID of the NEW SLIC matches the OLD SLIC, then no ACPI tables or OEM/Table ID strings are altered.

Note – this overrides any other options involved in table modification and OEM/Table ID string modification

Note – If the OEM/Table IDs of the SLICs don't match then this option is ignored

Only alter RSDT and XSDT tables

Only the RSDT and XSDT tables will have their OEM and Table IDs updated with that of the new SLIC table.

Note – 'Only alter ACPI modules in the main (BIOSCODxx) module' applies to this option.

Only replace OEM ID in additional tables

When any non-RSDT/XSDT tables are updated; only the OEM ID will be updated, not the Table ID.

Only alter tables in the main (BIOSCODxx) module

Only tables found in the main ACPI (BIOSCODxx) module will be updated. Exactly which ones and how they are updated depends on the state of 'Only alter RSDT and XSDT tables' and 'Only replace OEM ID in additional tables'.

Note – if this is ticked then other options are also influenced but all BIOSCODxx modules (not just the main ACPI containing module) are altered by these options. It is only ACPI tables that are only altered if they occur in the one main BIOSCODxx module.

Scan ACPI modules for OEM/Table IDs

By default all OEM/Table IDs found in the main ACPI (BIOSCODxx) module can be replaced (depending on later options). This option also scans all ACPIxx modules for additional OEM/Table IDs.

Note – ticking this option does not alter anything in the modules directly. It just influences other options.

Replace additional OEM/Table ID

Allows two user specified OEM/Table ID (14 character) or Table ID (8 character) strings.

Note – ticking this option does not alter anything in the modules directly. It just influences other options.

Note – 'Only replace complete OEM/Table IDs preceded by C3h' does not apply to user specified IDs.

Replace all OEM/Table ID occurrences

Replace any occurrence of the identified OEM/Table IDs. See 'Scan ACPI modules for OEM/Table IDs' and 'Replace additional OEM/Table ID' which describe how possible OEM/Table ID strings are identified. If this is unticked then no OEM/Table ID strings, either complete or split, are changed, other than those changed as part of the ACPI tables in the main ACPI (BIOSCODxx) module.

Note - if 'Only alter tables in the main (BIOSCODxx) module' is ticked then OEM/Table ID strings will only be replaced in BIOSCODxx modules.

Note - if 'Don't alter any ACPI tables' is ticked then any OEM/Table ID found in a valid ACPI table is ignored.

Note – If the OEM/Table ID is found to be in a valid ACPI table then the OEM/Table ID is altered respecting 'Only alter RSDT and XSDT tables' and 'Only replace OEM ID in additional tables'.

Only replace complete OEM/Table IDs preceded by C3h

Only replace complete (non-split) OEM/Table ID strings that are NOT part of a valid ACPI table (see above for these) that are preceded by a C3h byte. See 'Scan ACPI modules for OEM/Table IDs' and 'Replace additional OEM/Table ID' which describe how possible OEM/Table ID strings are identified.

Note - if 'Only alter tables in the main (BIOSCODxx) module' is ticked then OEM/Table ID strings will only be replaced in BIOSCODxx modules.

Note – this does NOT apply to 1) User specified additional OEM/Table IDs, 2) OEM/Table IDs from the RSDT/XSDT tables in a RW Everything report (it DOES apply to Table IDs only from the RW Everything report) and 3) Split OEM/Table IDs and split Table IDs.

Replace split OEM/Table IDs

OEM/Table ID strings that are split 4+2+4+4 are replaced. See 'Scan ACPI modules for OEM/Table IDs' and 'Replace additional OEM/Table ID' which describe how possible OEM/Table ID strings are identified.

Note – 'Only replace complete OEM/Table IDs preceded by C3h' does not apply.

Replace split Table IDs

Table ID strings that are split 4+4 are replaced. See 'Scan ACPI modules for OEM/Table IDs' and 'Replace additional OEM/Table ID' which describe how possible Table ID strings are identified.

Note – 'Only replace complete OEM/Table IDs preceded by C3h' does not apply.

Replace empty modules

Replace any appropriately sized and empty modules with the corresponding SLIC element. Currently will look for 374 bytes (full SLIC table), 338 bytes (pubkey and marker), 156 bytes (pubkey) and 182 bytes (marker).

Only replace SLIC elements that appear in the RW Everything report

Only SLIC elements that match the SLIC table from the RW Everything report are replaced. This does NOT include SLIC headers that are part of a full SLIC table or that are adjacent to a pubkey or marker (as the header can be modified by code and hence would not be expected to match). SLIC headers that are not in isolation are not matched but will be replaced if the other element(s) match, ie. If a non-matching SLIC header is part of a full SLIC table whose pubkey and marker match the RW Everything report, then the whole SLIC table, including the header, is replaced. All isolated SLIC headers will be replaced irrespective of matching (for the reasons above), but the log will only indicate a match if they do indeed match exactly.

Note – If there is no SLIC table in the RW Everything report then NOTHING will be replaced when this option is ticked.

Note – If there is no valid RW Everything report then this option is ignored.

Note – this is only active if ‘Replace existing SLIC elements’ for SSV2 mods.

Only replace OEM/Table IDs from RSDT/XSDT in the RW Everything report

Only the OEM/Table ID(s) that appear in the RSDT and XSDT tables in the RW Everything report will be replaced. This applies to those in ACPI tables and also those found in other modules. Note that all the other options still apply (ie. ACPI table selection options, preceding C3h option and split IDs options).

Note – If there is no valid RW Everything report then NO OEM/Table IDs will be replaced.

Note – If ‘Blank other ACPI table OEM/Table IDs’ is selected then the OEM/Table ID in all applicable ACPI tables will be blanked, ignoring this option.

Replace Table ID from RSDT/XSDT tables in the RW Everything report

Treat the Table ID from the RSDT and XSDT tables as strings to be altered in their own right (eg. for HP BIOSes with Capell00).

Note – ‘Only replace complete OEM/Table IDs preceded by C3h’ does apply.

Note – all relevant options above still apply.

Scan BIOSCOD modules for LENOVO IDs

Scan the BIOSCODx.ROM modules for extra LENOVO OEM/Table IDs.

Note – ‘Only replace complete OEM/Table IDs preceded by C3h’ does NOT apply.

Note – This options is selected by default if manufacturer is Lenovo

Note – This option is ignored with a valid RW Everything is parsed

Remove SONY OEM/Table ID lock

Removes the Sony OEM/Table ID lock (found in some Sony BIOSes ending in J6).

SSV2 Options

Replace existing SLIC elements

All identified SLIC elements, wherever they occur, are replaced with the corresponding element from the new SLIC. This includes headers, pubkeys and markers (and any combinations).

Note – this always happens in dynamic mods.

Note – this option is affected by the 'Only replace SLIC elements that appear in the RW Everything report' option

Replace existing SLIC table ID with []

The 'SLIC' string in ANY existing SLIC element in ANY module will be replaced with the specified 4 character string (default 'OEMS')

Note – this is not influenced by any of the above options. ALL modules are included

Note – this is NEVER performed for a dynamic mod

Only copy SLIC header OEM/Table ID

When inserting a full SLIC table, or SLIC header, only the OEM and Table IDs are copied; the remainder of the header is left intact.

Replace all 'SLIC' occurrences

Replace any occurrence of 'SLIC' string, that does not occur in an identified OEM/Table ID, with [] (see above).

Note – this is NEVER performed for a dynamic mod

Note - if 'Only alter tables in the main (BIOSCODxx) module' is ticked then 'SLIC' will only be replaced in BIOSCODxx modules AND in MOD_xxyy modules.

Only insert SLIC module

Does not attempt to find and patch ACPI modules. Will replace OEM/Table ID strings, SLIC elements and 'SLIC' strings.

Note – this is NEVER performed for a dynamic mod

Blank other ACPI table OEM/Table IDs

If ticked then the OEM/Table ID in ACPI tables, other than RSDT,XSDT and SLIC, are replaced with spaces. This aids module resizing.

Replace specific strings

Replace manufacturer specific strings to try and reduce module size. This is attempted after creator IDs are blanked and appropriate OEM/Table IDs are blanked, if 'Blank other ACPI table OEM/Table IDs' is selected. For example, 'Sony Corporation'.

Allow manual choice of SLIC location

Manually select the location, from a list of candidate locations, to insert the SLIC ACPI module. The tool will automatically scan for all valid locations before and after modules.

Manual location for SLIC insertion

The SLIC table will be inserted at this (hex) address in the ROM image. The tool will still check that it is valid (ie. not in a module or ROM hole).

Dynamic Options

Use SSV2 Method

If ticked an SSV2 style mod is performed (and all the options for SSV2, other than those noted, apply). If unticked then Prepare/Catenate are used.

Control Options

Ask prior to each modification

A dialog box allows the user to choose whether each modification to a module or the BIOS image is implemented.

Allow user modification of modules

If the tool fails to resize a module automatically the user is given the opportunity to modify the module manually, the tool then has one further attempt (without any additional modification) to resize the module.

Always allow user modification of modules

The user is always given the opportunity to modify the modules manually, irrespective of sizing success, the tool then confirms the new module will fit.

Allow user to modify other modules

Pause to allow user to modify any module in the DUMP directory.

Process all compress modules (EFI)

Process all compressed modules to see if they contain other modules or ROM images.

Extract modules when verifying

When verifying the BIOS, extract the modules as well.

Use 'brute force' search for compressed images

Rather than parsing the modules in compressed images in turn; scan the entire image for any valid modules irrespective of location.

General Notes

Phoenix OEM/Table ID handling

SCANNING - For all BIOSes these ALWAYS occur:

- the main BIOSCODxx module (containing the RSDT table) is scanned for valid ACPI tables
- all ACPI tables found are scanned for OEM/Table IDs
- any existing SLIC tables/headers/markers are scanned for OEM/Table IDs

EXTRAS:

- if the 'scan ACPI modules for OEM/Table IDs' is ticked then all valid ACPI modules in ANY module are scanned for OEM/Table IDs
- any additional user specified OEM/Table IDs are added
- If there is a RW report then every table in the report is scanned for OEM/Table IDs
- If 'Replace Table ID from RSDT/XSDT tables in the RW Everything report' is selected then the Table ID from RSDT and XSDT are included in their own right
- If 'Scan BIOSCOD modules for LENOVO IDs' and there is no RW report then the BIOSCODx.ROM modules are scanned for Lenovo OEM/Table IDs

The list can be seen in the log:

OEM/Table IDs identified are:

1. FSC PC
2. FUJ FJNB1AF
3. FUJ PC
4. DOCK FJNB1AF
5. CPPM CpuPm
6. CP0I Cpu0Ist
7. CP1I Cpu1Ist
8. CP0C Cpu0Cst
9. CP1C Cpu1Cst
10. AHCI FJNB1AF
11. IDE FJNB1AF
12. VISTA FJNB1AF

Then, depending on the selected options, occurrences of these OEM/Table IDs are changed to the one from the new SLIC table.

By default this includes:

- Those in ACPI tables in the main BIOSCODxx module (depends on the 'Don't alter any ACPI tables' option, the 'Only alter RSDT and XSDT tables' option and the 'Only replace OEM ID in additional tables') are changed
- Stand alone strings are replaced dependent upon the 'Only replace complete OEM/Table IDs preceded by C3h' - by default those preceded by a C3h byte are replaced (option is ticked)
- The C3h option does NOT apply to User specified additional OEM/Table IDs, OEM/Table IDs from the RSDT/XSDT tables in a RW Everything report (it DOES apply to Table IDs only from the RW Everything report) and Split OEM/Table IDs and split Table IDs
- Split (in 4+2+4+4 pattern) OEM/Table IDs are replaced dependent on the 'Replace split OEM/Table IDs' option - by default this is selected

EXTRAS:

- If 'Only alter tables in the main ACPI (BIOSCODxx) module' is UNTICKED then all ACPI tables in all modules have their OEM/Table ID (or just OEM ID depending on the above options) changed.
- If 'Replace split Table IDs' is ticked then isolated split (in a 4+4 pattern) table IDs are replaced.

NOTE:

- if 'Only replace OEM/Table IDs from RSDT/XSDT in the RW Everything report' is selected then only these IDs are replaced

Handling of SLIC elements

A SLIC element is considered to be a SLIC header, pubkey or marker (or any combination thereof). A full SLIC table includes 3 elements – the header, the pubkey and the marker.

With NO RW report

- SSV2 – No existing elements are replaced. A new SLIC table is added.
- Dynamic - ALL existing elements are replaced.

With RW report

The tool makes the following 3 decisions sequentially:

- 1) If there is an EXISTING whole SLIC in the BIOS AND a whole SLIC in the RW report then dynamic is selected with ONLY the elements matching those in the RW report being replaced in the BIOS (unless any element is in the bootblock in which case SSV2 is selected).
- 2) If there are ANY SLIC elements in the RW report then SSV2 is selected WITH matching elements from RW report replaced in the BIOS.
- 3) If there are NO SLIC elements at all in the RW report then SSV2 is selected with nothing replaced (as above).