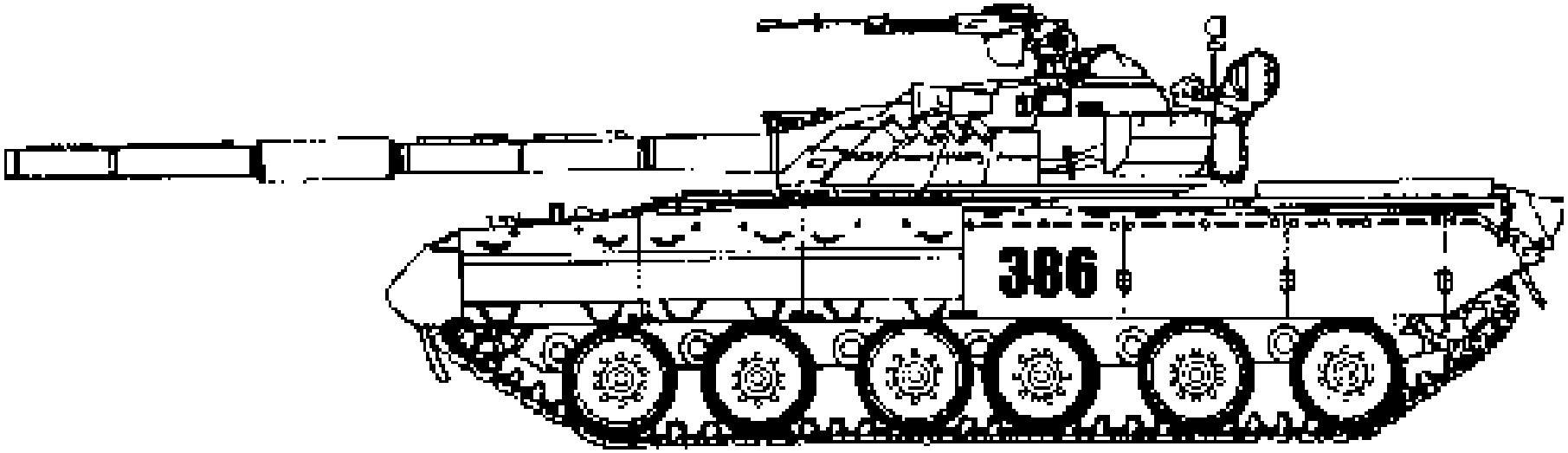


80_PA

alternative (hacker) implementation of technology of online activation of SecuROM PA (Sony DADC AG)

« Tiberiumny reversing»

2011-2026

«When piracy gets the best of DRM, and pirates are much stronger than people who fight with them. Yokhokho!»

Table of contents

1. **80_PA. Important information. Responses to questions** 4
2. **80_PA. Short description of technology of SecuROM PA. Terminology** 8
3. **80_PA. Procedure of generation of the correct unlock code. Description of possible errors and their elimination** 12
4. **80_PA. Review of windows** 29
5. **80_PA. Android com.lab_80pa** 38
6. **80_PA. MacOSX «Cider» / Linux Wine** 39
7. **80_PA. Disclosure structures of unlock (request) code** 40
8. **80_PA. Other projects** 48
9. **80_PA. About the project 80_PA. Feedback** 51

original hashes

V 1.1:	***MD5:*** 8a4d3601e76c6fbfabed3f72695fd042	***SHA1:*** ae30188a993d26b3e442aee58df6db02b5934fa2
V 1.2:	***MD5:*** e30aa4d4615b2c237a63d06039f7fb12	***SHA1:*** 787c8f7a90c23d77a0ade0e781926dee730c9539
V 1.2.1:	***MD5:*** 7371519521cd3f18a097c9160aed48b1	***SHA1:*** d9dcf7add7b69c0fdc5af0473a52bfe8a58071c1
V 1.2.1 hotfix:	***MD5:*** b67a7d6050f5ecc727c89d925a2d6b69	***SHA1:*** ddafd26396b18cfd66a96e2273f0803bb326766d
V 1.2.2 hotfix:	***MD5:*** c06535fee8af5620f7023e2ff91c7f79	***SHA1:*** 7a8b8d1c92477fd789acce34bc9253c659c3678c
V 1.2.3 hotfix («Chinese edition»):	***MD5:*** 3622da94e73da6f74ccc1c02e80b6aa4	***SHA1:*** 8cb081cdeac02a03697ba52df0fa1a4bca451b93
V 1.3.0 big update:	***MD5:*** ff33b0d5ae28f0f90e75300929d1ce68	***SHA1:*** b61f483d12c856f3916091cbeca273b0f27d219d
V 1.3.1 hotfix («Telltale edition»):	***MD5:*** db2cc965529de727e544291a0aa69004	***SHA1:*** 2b3f08c52b4033caf4c7e96db6179ba55466416a
V 1.3.2 hotfix («over 100 key kits...»):	***MD5:*** b46460557081218742478f8d5585e995	***SHA1:*** 90ca6fe4620a6b03f2a9dc56d02a0c93eeb505a7
V 1.3.3 hotfix («"review update...»):	***MD5:*** 91510d698fb57b870acef168d668119d	***SHA1:*** fe139441f8bece2731439462dfc9c957da9a7031
V 1.3.4 hotfix («"kav update...»):	***MD5:*** 43a34a79764c2fe0069fb23041ce0781	***SHA1:*** 43d3f187006a7a49adfa05876be7cea886b072e5
V 1.3.5 hotfix («"five»):	***MD5:*** 90aa774709a255f47ea1ea908b521f1c	***SHA1:*** 5759cc7a3db467157b2e33cf07f805555a75a364
V 2.0 («2020»):	***MD5:*** 78259f563deb8b857cd63c1d4e08c010	***SHA1:*** 3bbde1fbe3865cf2e89ee83a56932e3c46485271
V 2.0 hotfix («2020»):	***MD5:*** 71040679a0a0a85ca12f4bfb0edbb3fe	***SHA1:*** fe4125af9e378caea4b8f4bf921a6be6d12bbebd
V 2.0 new («2022»):	***MD5:*** D7E142142D470E5FC9642BC3253FD612	***SHA1:*** 72326EF9AC18B60F2B3BD32A37D2679A58AC56EC
V 2.0.3 («2023»):	***MD5:*** 4B44BA20D5DA3814E47A96CA7CFC7B2C	***SHA1:*** 766331908D00D3B10A42E9BEB2AA173BFA67DF3B
V 2.1.0 («2023_761»):	***MD5:*** 98AAE36593F3D578C81985099C4ACB4F	***SHA1:*** 070C2AE1235A6075B4A2D8801E7ED14DCB363858
V 2.7.0 («07/2023»):	***MD5:*** D21356EFCDD798B76B1EAD4CBD0A87DD	***SHA1:*** ED6D8E36E1709596E4E70B12260FE3B3D5733727
V 2.7.1 («10/2023» - 600):	***MD5:*** 742B60209D6FC8CA530A586507DC6B25	***SHA1:*** 73CDE0FADCC28DE692C58FA02D1653271CB8E39D
V 2.7.2 («04/2024» - rampage!!!):	***MD5:*** 7662FED60C37541A8D0EE7EF31B7D9AF	***SHA1:*** 79584814FC935E7071C5CDB70DCC62FF8651DA4C
V 2.7.3 («07/2024» - rampage2!!!):	***MD5:*** 300EA5353106F1CF868DFA55808EC8DD	***SHA1:*** 32FBC21AAF02575A005411EDDE234607C25DCE4D
V 2.7.5 («03/2025» -yates edition):	***MD5:*** 07864AC225831229C904C4B020043B37	***SHA1:*** 8A50634BE26A19DD0026873A0992472D9BF005A0
V 2.7.7 («01/2026» -CRISTRMAS):	***MD5:*** FA0DECA6D382EC78D7FAEBF9BC80C325	***SHA1:*** 75408381FF298CE7C0DA57961C4A2915E2723B01

Important information. Responses to questions

Why it is necessary 80_PA? First of all, to mock at Sony DADC AG and personally Reyngard Blaukovich. And in essence: if you use license copies of games with SecuROM PA, you can, absolutely legally, using *80_PA*, to register these games bypassing serial number (s/n), official without knowledge. And here games with the SecuROM «Trial mode» (for example, from *«Big Fish Games»*) and *EA Game Authorization Management*, get where, by means of simple shifts, to force SecuROM to be activated manually (Manual activation).

Regarding distribution of an executable file and provision of source codes. The source file *80_PA.exe* it is possible and it is necessary to spread on any Internet resources, without forgetting to fit this instruction on use also. Specifying of a reference to the source https://exelab.ru/f/PAUnlock is welcomed. The original source code 80_PA (except function of generation of SecuROM HWID and some auxiliary) doesn't extend publicly yet and is available at some circle of interested persons. Sale and thrashing of copyrights aren't allowed. If you have a desire to deal with technology of online activation of SecuROM – WRITE to the AUTHOR of the PROGRAM!

Regarding harmful maintenance of an executable file. Download «80_PA.exe» only from the trusted sources (exelab.ru, cracklab.ru, antistarforce.com, rutracker.org, ~~securom.com~~, ~~denuvo.com~~)! The soriginal file 80_PA.exe NOT contains any destructive and malicious code and can't do harm to your computer. The vast majority of a code is the cryptography operations taken from the OpenSSL (https://www.openssl.org/source/) and BigDigits (http://www.di-mgt.com.au/bigdigits.html). Remaining information was received in the way a reverse engineering of the original SecuROM PA technology from the protected games (*Epic Mickey 2: The Power of Two, Grand Theft Auto IV, Bioshock, etc.*). Protection *VMProtect* is hung up specially to minimize appearance of the «untrusted» modified copies of the original program.

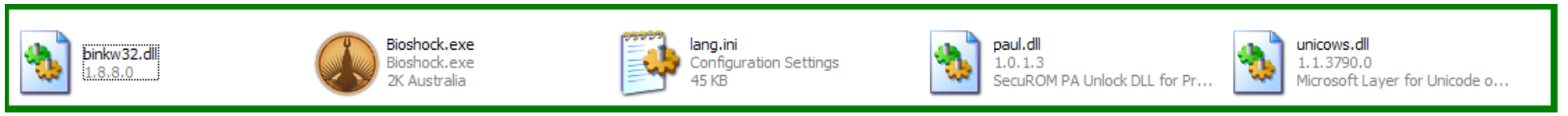
If the generated unlock code isn't accepted by SecuROM. Anything terrible isn't present! In 99,99999% cases the problem in a hash of serial number which was already used and is stored in the register at SecuROM. For deleting the used hashes it will be required to clean a certain branch of the register or it is possible to come on the other hand – simply to set in expanded options (*[80_PA] Advanced*) other *UC.Serial number stamp* value in the Hex format (2 bytes). In more detail in point 3.

If my game isn't present in the «*Aviable KEY KITs*» list. Unfortunately, by the time of release 80_PA it wasn't succeeded to collect full basis of the protected games though common efforts succeeded to get such rare games as *ys7* (*ys seven*). But if your toy using the SecuROM PA technology is absent in the list, you can help to add it! Pass in a directory of installation of a game, and collect min. work set which shall include:

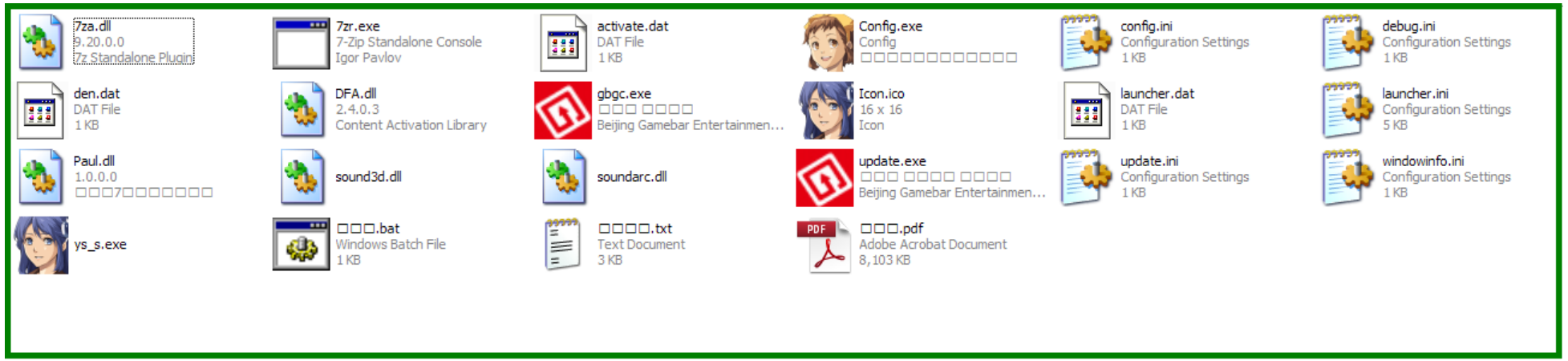
- The main .exe the file which is protected by SecuROM PA;
- **PAUL.DLL**, **dfa**, **lang.ini** *(last two if is)*;
- All possible auxiliary dynamic .dll *(for example,* binkw32.dll*)* in this directory and .exe files;
- Different small-sized .INI, .txt, and .dat files;

The exception is made by games from the «Telltale Games» company: the original installer of a game is required (for example, *Bone_Out_From_Boneville_Setup.exe*), in view of the specific interface created for SecuROM PA.

Example #1. BioShock:



Example #2. Ys Seven:



Having aggregated the specified files, archive in archive of the .zip format or.7z (in the amount over 30 Megabytes shan't turn out) flood on a file hosting service (www. wetransfer.com is recommended). Send the link to us by mail (it is specified in contacts) or on the site exelab.ru (cracklab.team). Sets of cryptographic keys will be torn out and added to library 80_PA!

A little bit long there is a generation of unlock code and very long there is a decoding of unlock request code. ~~Yes! ☺ We faked a little and didn't begin to tear out static DES a key in both cases, having chosen the line of least resistance. However, if someone is ready to make it, let know! Your name will be entered in the list.~~ Already fixed in v.2.0 (2020-2023)

Are the SecuROM and DENUVO source codes sale is ACTUAL? Naturally, more than ever before. Contact me by all available means by writing private message to cracklab.team (exelab.ru) in advance. qTox - the most preferred option! *Telegram* is also available. The source code is needed only for internal research by one person only (i).

Forced switching on of manual (Manual) activation for games with Trial-mode («*BigFish Games*») with reset of locks.

Epic trick:

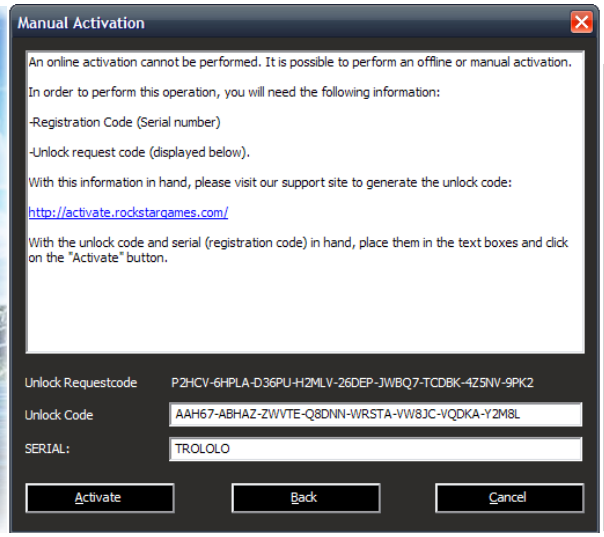
1. **We replace the current new paul.dll version (normally v2.x) in the directory of a game with the ancient paul.dll version (v 1.x)**
2. **We have an opportunity of «Manual activation»!**
3. **We use 80_PA**
4. **We drop all three LOCK bits (on a default, shall be dropped) in the service structure**
5. **We generate the free unlock code**
6. **We copy-paste and activate**
7. **PROFIT!!!**

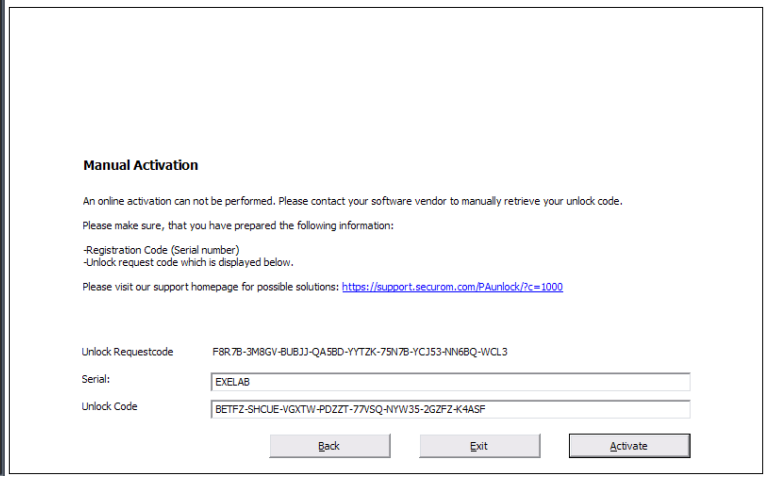
Forced deleting HKEY_CURRENT_USER\Software\SecuROM\License information and! CAUTION!... Simply press the «Hidden reg keys» button. A trick essence in use of undocumented opportunities of reading/creation of branches of the register by means of low-level functions from ntdll.dll and the accounting of null-byte at the end of a branch name.

About DENUVO and final cracking of SecuROM. The project 80_PA is not the single achievement in case of the research SecuROM. Upon, almost all opportunities of SecuROM were probed and cracked: beginning from banal anti-debugging and finishing with the virtual machine (VM) with the module of check of compact disks. Critical vulnerability which affects all versions of protection was found in the last and allows to launch terribly the protected programs without original license compact disk and even without traditional Alcohol of 120% with Daemon Tools! (https://xakep.ru/2015/08/07/securom/). Also operation over DENUVO - a topic (https://exelab.ru/f/index.php?action=vthread&forum=13&topic=19719) is the most exact primary source of information on this protection (better, than at 3dm).

Short description of technology of SecuROM PA. Terminology.







Terminology (glossary):

SecuROM PA (Product activation, online-activation)	**Actually, original technology of online activation of SecuROM**
SONY DADC AG	**The company which made SecuROM**
HWID (Hardware ID)	**Unique identification number of your computer which is created of different data on the set hardware. Each protection creates it on the personal algorithm. SecuROM is regarding, the algorithm of generation will be described below.**
MD5 (Message Digest 5)	**The 128-bit hashing algorithm developed by professor Ronald L. Rivest from Massachusetts Institute of Technology**

	(Massachusetts Institute of Technology, MIT) in 1991. It is intended for creation of "prints" or digests of the message of arbitrary length and the subsequent check of their authenticity
DES (data encryption standard)	The algorithm for the symmetric encoding developed by IBM firm and approved by the U.S. Government in 1977 as the official standard (FIPS 46-3). The unit size for DES is equal 64 bits. Feystel's network with 16 cycles (rounds) and the key having length of 56 bits is the cornerstone of algorithm.
RSA (Rivest, Shamir и Adleman)	The cryptographic algorithm with public key which is based on computing complexity of the task of factorization of large integral numbers.
CRC (Cyclic redundancy check)	The cyclic redundancy code - the algorithm of finding of checksum intended for check of integrity of data.
XOR	Bit operation (excluding "OR").
appid	Unique identifier (3 lines * 16 bytes = 48 bytes) which SecuROM appropriates to any game.
Unlock requestcode	Code request on the SONY DADC AG server containing the ciphered HWID of your machine (RSA) and the service structure (DES) which contains also CRC from appid MD5 hash, for receiving unlock code.
Unlock code	The code response generated by the server, by data from unlock requestcode, but with other keys. In a code response there is the service structure (DES) and the ciphered HWID (RSA). In the service structure of unlock code the hash of serial number is considered.

Serial (s/n или serial number)	**Serial number which is written usually on the acquired license disk. For the server is the guarantor of that you are the buyer of a disk. In implementation 80_PA legally acquired serial number generally isn't required! Its digest will be generated off the wall or entered by you from a ceiling.**
47 (0x2f)	**unlock code length**
52 (0x34)	**unlock requestcode length**
48 (0x30)	**appid length**
28 (0x1C)	**HWID string(ASCII) length**
16 (0x10)	**HWID length in bytes**

All procedure of generation shares conditionally on three stages:

1. **Generation of HWID, formation of unlock requestcode by the user's machine with use of appid;**
2. **Sending unlock requestcode for the server. Decryption and check on the unlock requestcode server, on condition of finding of s/n in basis. Extraction of the digest of appid and other service data from requestcode, formation of unlockcode with use of other encrypting keys. Adding of LOCK bytes if it is required. Sending unlock code back on the user's machine;**
3. **Receiving unlock code. Decryption. A check of the digest of serial number with saved earlier. Extraction of HWID from unlockcode and generation of HWID by the current machine. A check of two received HWID on a mask.**
4. **it is conditionally possible to carry here. Check of HWID in case of each start.**
5. **HWID consists of hashes over which logical transaction of XOR is applied:**

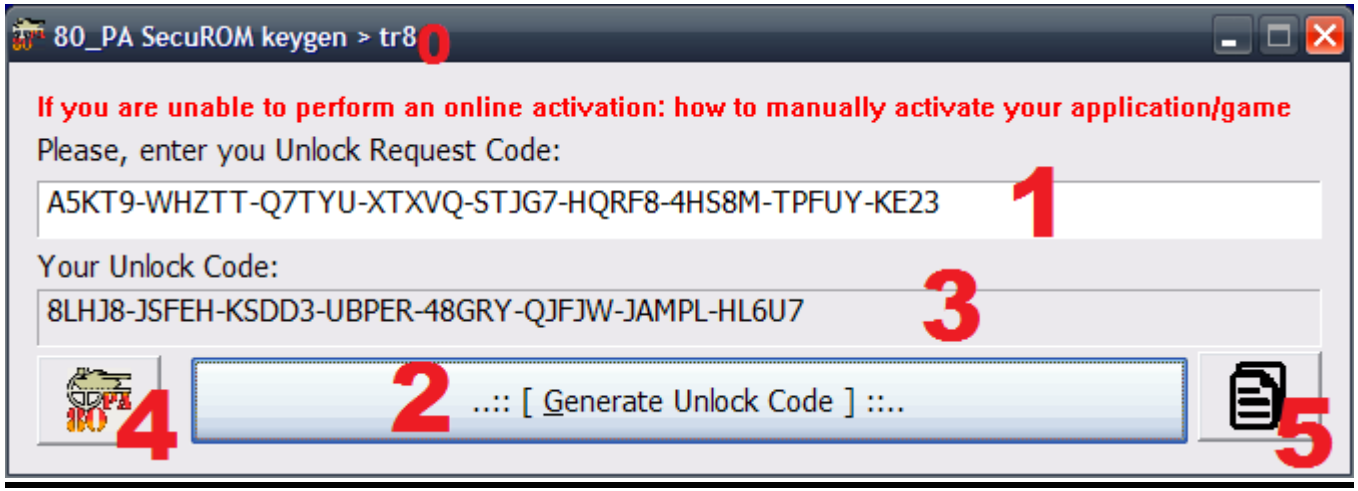
- **Information on an operating system (WINAPI kernel32.GetVersionEx)***
- **Information on the established processor (WINAPI kernel32.GetSystemInfo)***
- **Information on the established videocard (WINAPI d3d9.Direct3DCreate9)***
- **Information on the network interface card (WINAPI iphlpapi.GetAdaptersInfo)**
- **Information on serial number of system volume on which Windows is established (WINAPI kernel32.GetVolumeInformation)***
- **Information on remaining serial numbers of volumes (WINAPI kernel32.GetVolumeInformation)**

***According to a mask, SecuROM checks only the specified hashes.**

Beginning from Bioshock and finishing with the latest protected games, procedure of activation is identical byte in byte!!! Naturally, distinctions only in addresses, appid and special constants which are used for a check of results of operation of functions of online activation. HWID will be different by any machine. After change of your configuration of hardware (for example, you changed the video card), in case of next run of SecuROM will find mismatch of HWID and activation will be required again. Also, the hash of serial number will be skidded in «black list» which it is possible to clear or carry out an official response of a key (revoke) illegally.

Procedure of generation of the correct unlock code. Description of possible errors and their elimination.

version 2 (now is main)

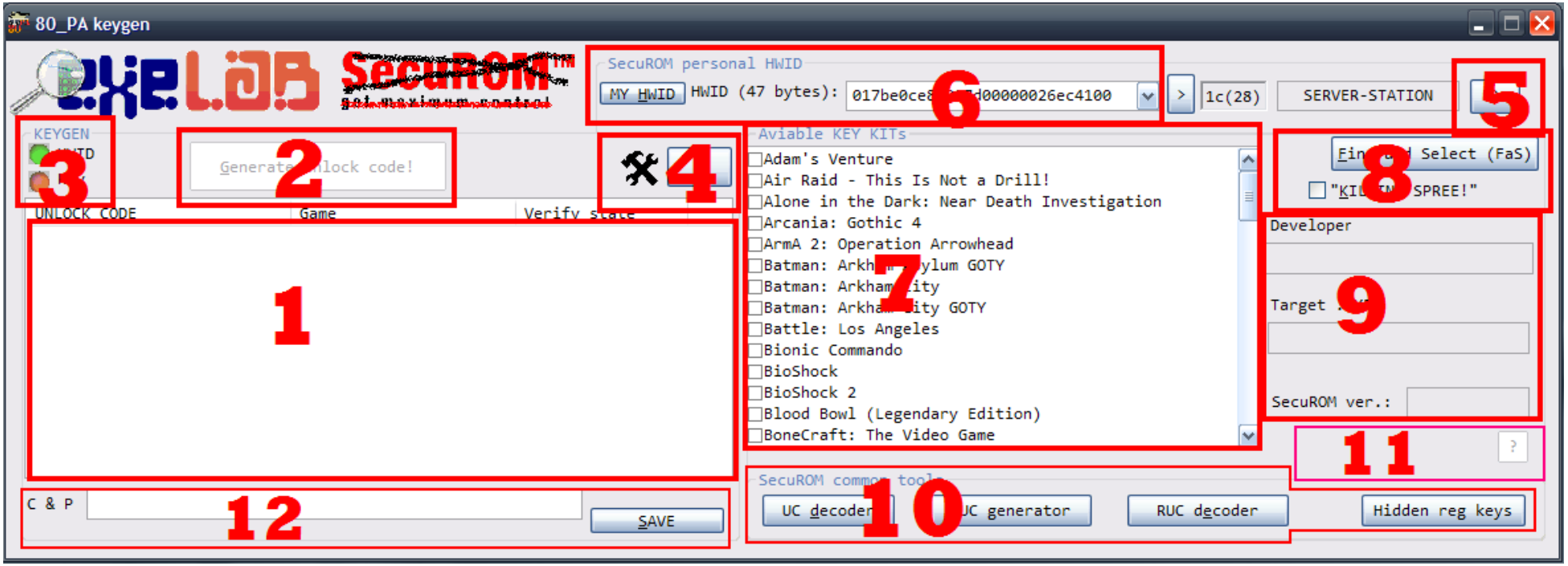


0. Already launched game with SecuROM 7-8 can be detect automatically (0). For example: Tomb Raider
1. Enter REQUEST CODE (request code, longer than the response code) in field (1)
2. Click “Generate Unlock Code” (generate the response code) using the button (2)
3. We take ready Unlock Code from the input field (3)
4. If necessary, call the previous extended version of 80_PA by clicking on the button (4)
5. If you not want copy with standard keyboard configuration (Ctrl+C), fast click on the button (5)

6. The transition is carried out from the main window by pressing the button at the bottom left or by calling 80_PA.exe with any arguments.

7. The first version of the program is as follows:

version 1 (secondary is now)



Legend:

(1) ListBox in which the generated unlock code are displayed;

(2) The «Generate unlock code button» which actually launches generation process;

(3) Control lamps: a validity of HWID and the selected key set (games);

(4) Expanded options of generation (key values of service part of unlock code, cryptography parameters)

(5) Expanded information on SecuROM HWID

(6) The current SecuROM HWID value (initially corresponds to your personal HWID). It is possible to change at discretion.

(7) Available sets of cryptographic keys (appid, DES, RSA) for generation – the list of games.

(8) The option "Find and Select" – looks for in the launched processes on file name of game KEY KITs with the SecuROM PA technology. Considerably simplifies finding of the correct key set for generation. Nearby there is CheckBox "Killing spree!"– selects all available key sets (games). Repeated clicking removes separation.

(9) Information on the selected game (the developer, a name of the target .exe file, the version of SecuROM)

(10) «UC Decoder» - decoder unlock code. You can choose to check the structure of the generated unlock code. «RUC Generator» - generator request unlock code. We need to form a fictitious request unlock code when sending the request to the official activation server SecuROM PA. «RUC Decoder» - Decoder request unlock code. «Hidden reg keys» - allows viewing and deleting inaccessible reg branches *\HKCU\SOFTWARE\SecuROM\License information* и *\HKCU\SOFTWARE\SecuROM\!CAUTION! NEVER DELETE OR CHANGE ANY KEY*

(11) Panel of icons. Highlights important notes on a game (regarding requirements of changeover of library-wrapper paul.dll and lang.ini from the enclosed archives in the «80_PA addons folder» of an official set of a keygen 80_PA, and also information on possible use of «EA Game Authorization Management» technology). Detail information can be obtained, having clicked on the button "?" close to icons

(12) The last generated unlock code will be inserted into TextBox. You can also click on any generated unlock code in ListBox (specified in point 1), for convenient copying of the line unlock code. The «Save» button will create the report from all generated unlock code and will save it on a disk in any place specified by you.

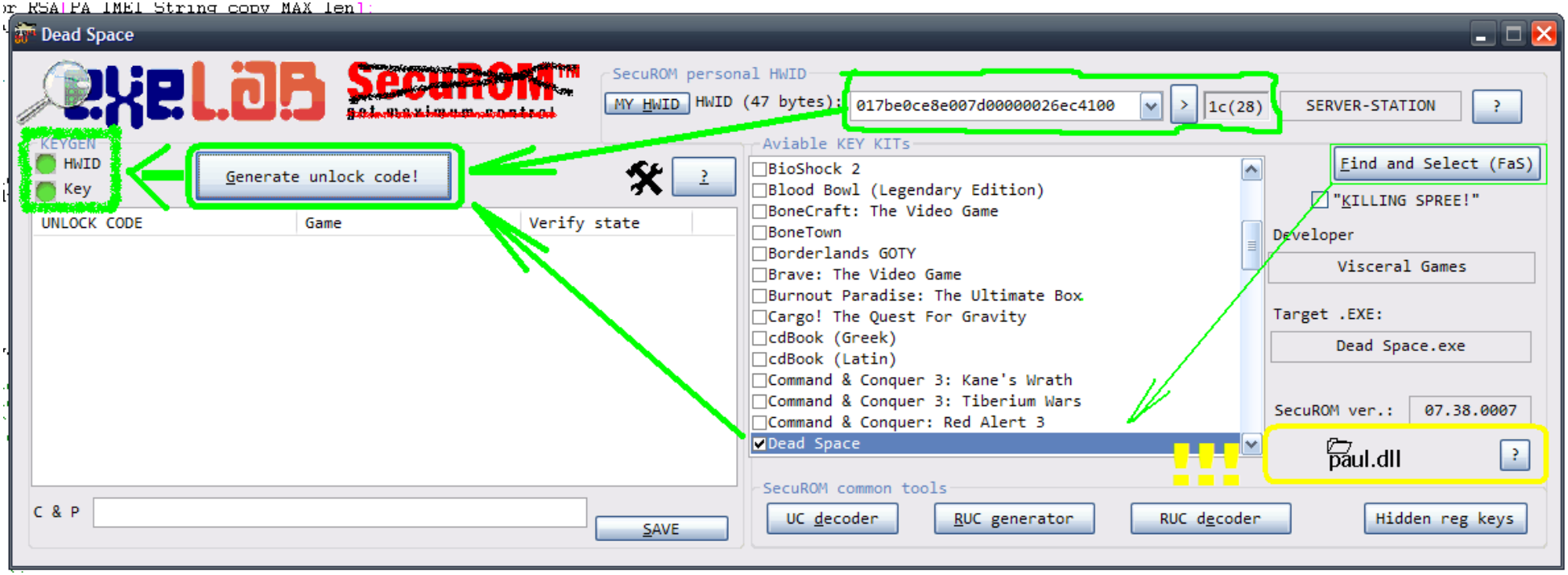
8. We decide on a game for which it is necessary to generate unlock code. There are three options:

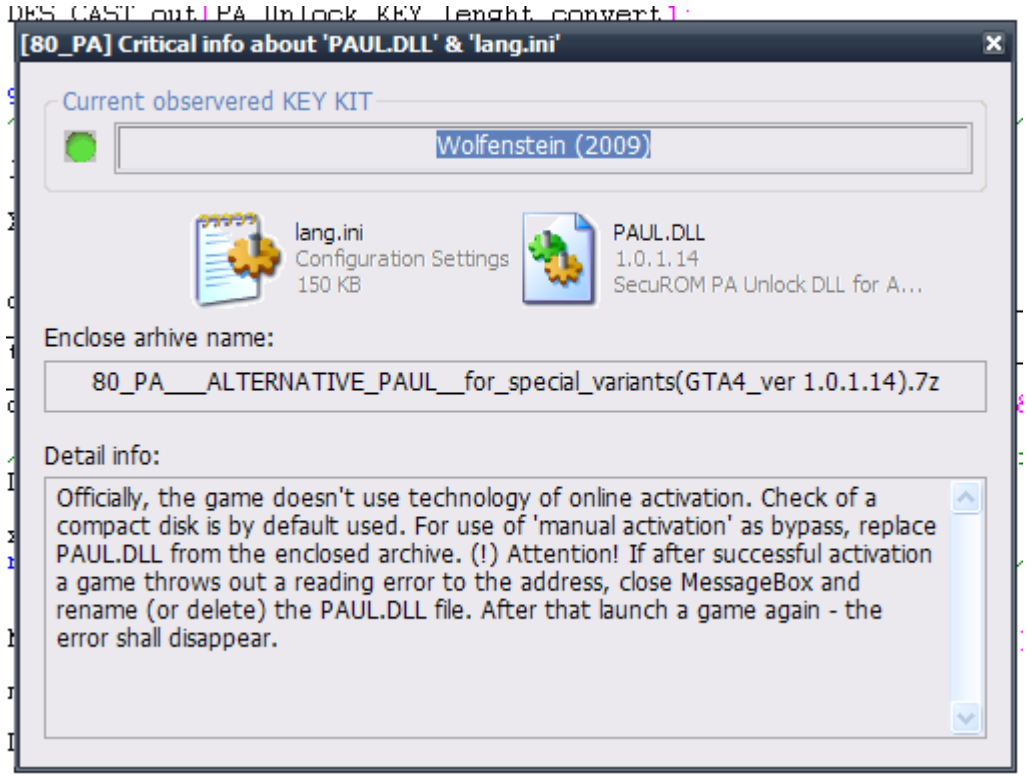
a) We select the required quantity of games manually from the list (7) – are ticked off;

b) If game is launched and SecuROM manual-activation is active, then simply we involve option «FaS» (8). In a cap of a primary window the name of the selected game will be displayed thus or will be otherwise specified that is found nothing («FaS – Nothing found»)

c) We select «Killing spree!» option, to generate unlock code for all available games in library 80_PA;

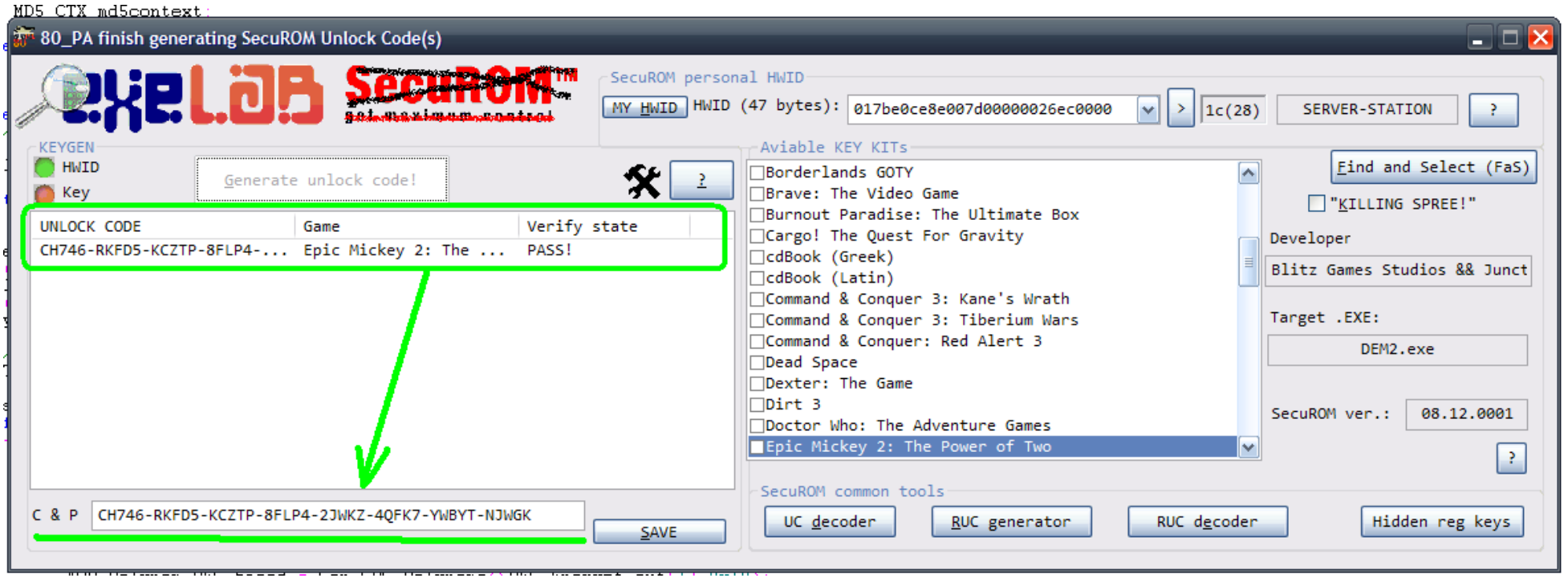
9. We are convinced that all two control lamps (3) burn in the green color. It is specifying for an unblocking of the button of start of generation (2). Pay attention also to the icons which are outlined in yellow color – it is possible, you need to execute additional operations with the paul.dll (Product Activation Unlock Library. Dynamic Link Library) and lang.ini (Language) files. Archives (in a format .7z) with the called files are included in the official package 80_PA and are located in the folder with the name 80_PA addons



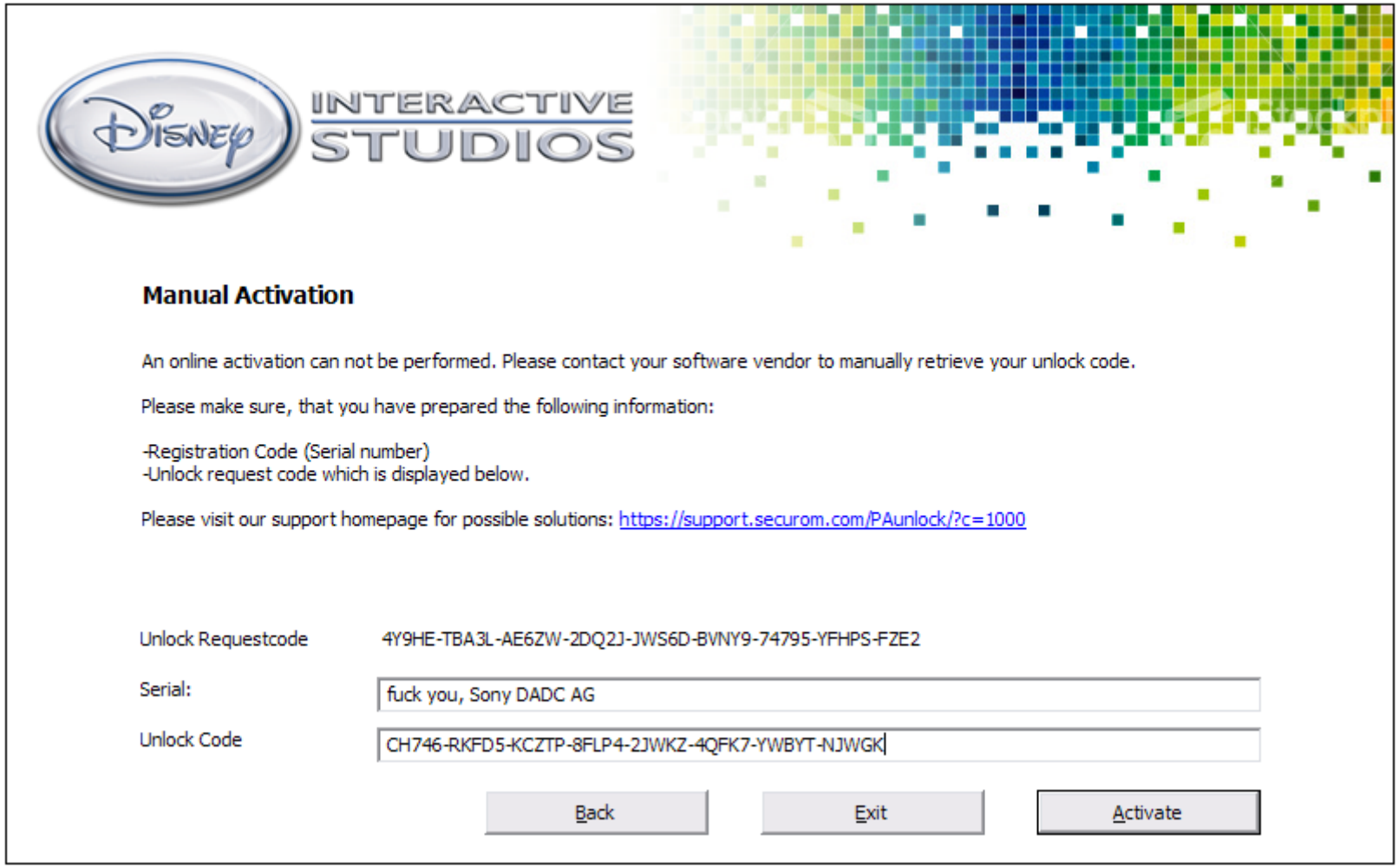


10. **We click (2), the cap of the main thing will accept the message «80_PA start generating SecuROM Unlock Code(s)» also we wait for some time (depending on the power of the processor and number of the selected games);**

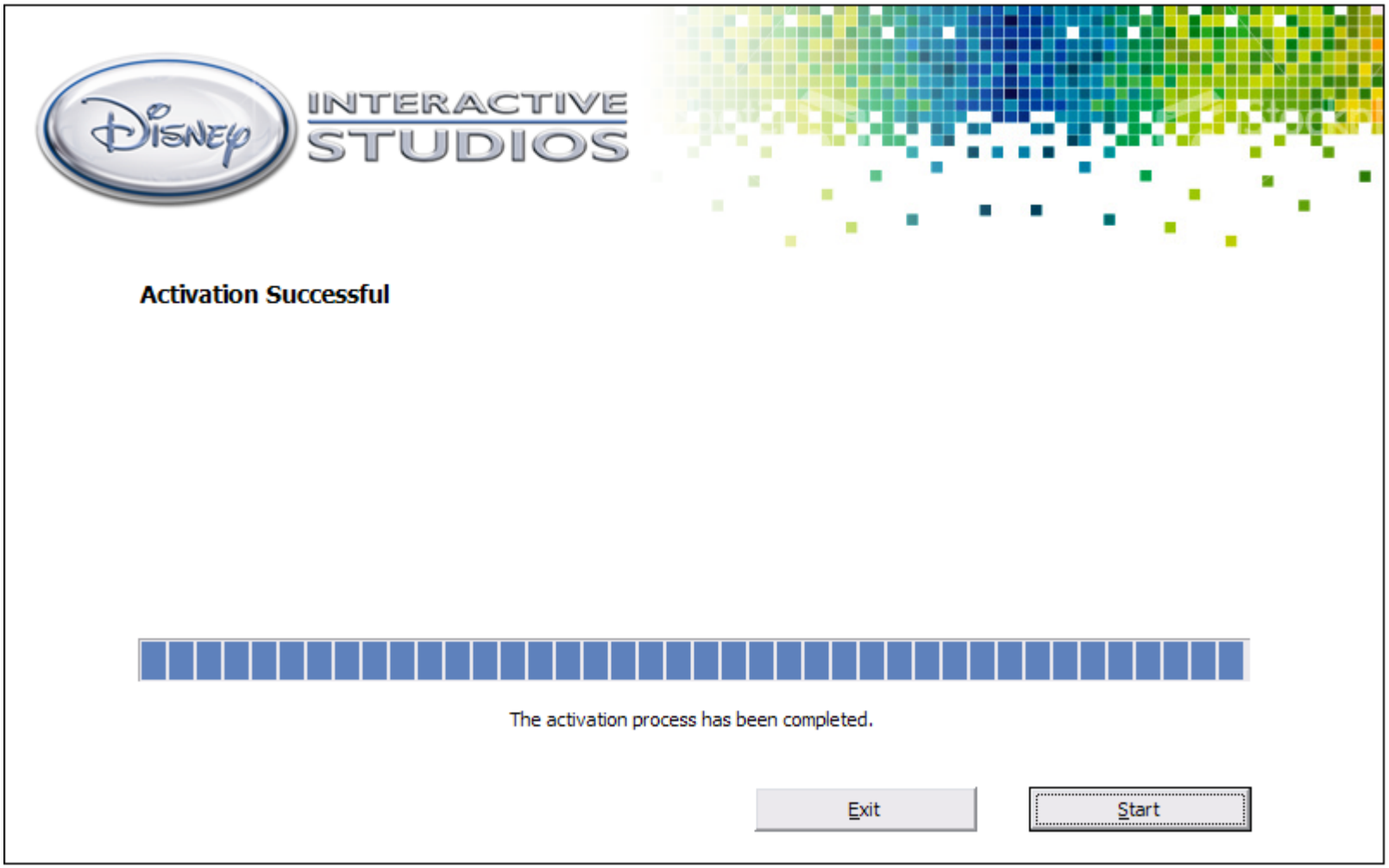
11. We wait for the end of generation of unlock code. However, in case of a multiple selection 80_PA will periodically add the generated data to the list, and they can use. The last generated unlock code is displayed in the EditBox data entry field **C & P** (Copy & Paste) from where it is possible to copy a code response without problems. It is possible to select a code response for copying from the list above, having right-clicked. The full end of generation will be marked by the message in a cap of a primary window «**80_PA finish generating SecuROM Unlock Code(s)**». The column «Verify state» (the status of check) displays result of check of unlock code on the algorithm put in the protected SecuROM PA files («**PASS!**» – check is taken place completely; «**Invalid HWID part**» - all two stages of unpacking of unlock code are passed, however the received HWID doesn't match HWID of your machine; «**UC not unpack**» - unlock code can't be unpacked at the first stage of receiving its official part)



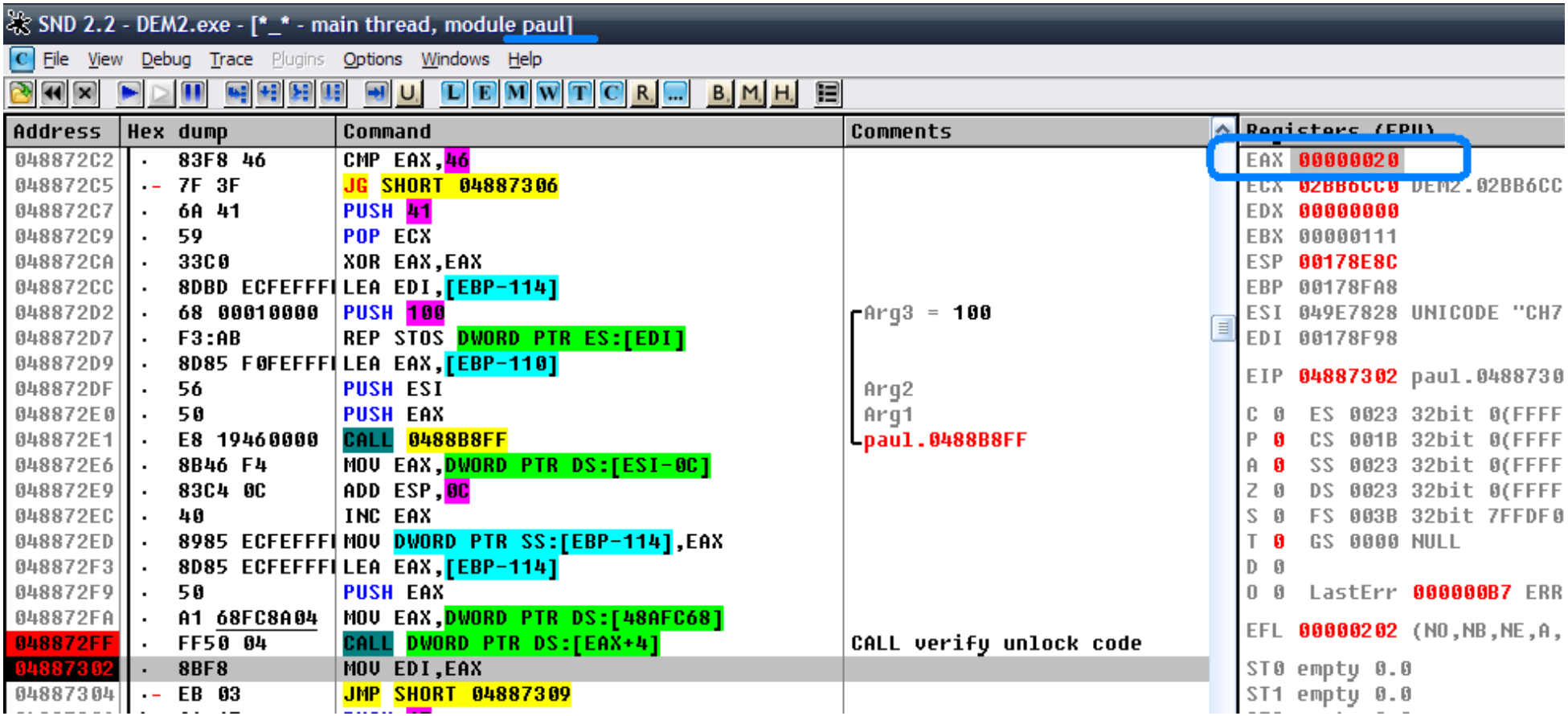
12. **We insert the generated unlock code into the appropriate data entry field of the Manual activation window. We enter any rubbish into the Serial data entry field (a hogwash, nonsense, a crap, from a ceiling, from a lamp, etc). We click «Activate»!**



13. If everything was made correctly, in ideal option you shall see «Activation Successful». You click «Start» and quietly you play … until when the license SecuROM PA can be lost because of installation of new hardware or a meeting of video drivers.



14. If activation came to the end with an error, and you read this point. Relevant for early versions of 80_PA, this is unrealistic in ver2.0. In the most extreme case, you need to generate the unlock code again. Nevertheless, first of all, do not panic! If you the advanced hacker, can precisely learn the error state number, having glanced in the program a debugger – is banal set the break point after a call dynamic library paul.dll of procedure of check of unlock code. In the 32nd bit register of the EAX processor the error state number will be displayed. We will allow in OllyDbg SND 2.2 it will look so:



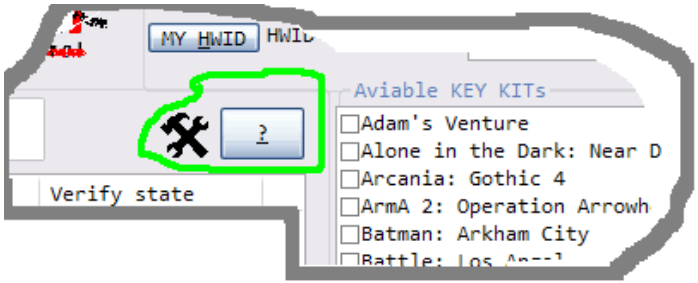
In this case the code 0x20 says that the digest of serial number (serial) is in a black list in local storage SecuROM PA. It is the most widespread error in activation process.

The table of the most often found codes of errors is given below.

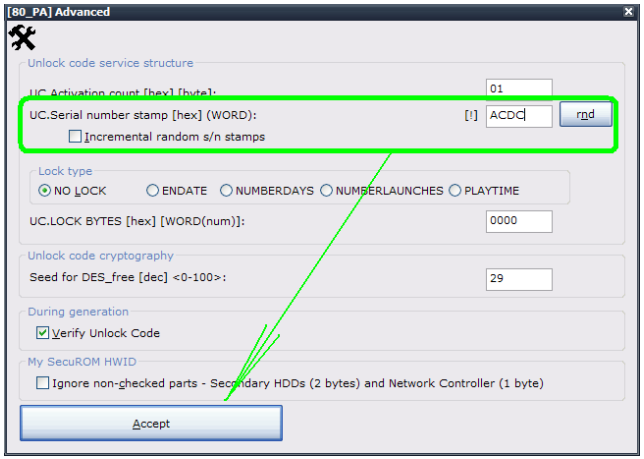
Often met codes of the errors returned by procedure of check of unlock code		
The error state number in the register EAX after a call **(HEX-format)**	The conditional macro / compliance in 80_PA	Description
1	**PA_ERROR_SUCCESS**	**Activation is carried out successfully**
7	PA_ERROR_UNLOCK_LEN_MISMATCH	**Length of the line unlock code isn't equal to 47 bytes**
9	PA_ERROR_IMEI_PART_NOT_VALIDATE **«Invalid HWID part»**	**It isn't unpacked HWID (IMEI) part of unlock code doesn't meet).** ***Probable cause:*** **the received HWID from unlock code doesn't match your machine.**
0x14	PA_ERROR_UNLOCK_SERVICE_PART_NOT_VALIDATE **«UC not unpack»**	**The service part of unlock code isn't unpacked.** ***Probable cause:*** **any seed from the range 0-100 isn't suitable for a randomny set of DES or created from personal appid, the digest doesn't match the registered digest in the service part of unlock code (unlock code are confused).**
0x20	**PA_ERROR_SERIAL_DIGEST_BLACK_LIST**	**The digest of serial number (serial) is in a black list in local storage SecuROM PA**

If you don't own technique of debugging, then anything terrible is also not present. In 99,9999% of cases the error will be PA_ERROR_SERIAL_DIGEST_BLACK_LIST (the digest of serial number locally is banned by protection). We will consider some candidate solutions of this misunderstanding:

I. **Most simple and fast, with use 80_PA.** Actually, the most obvious that it is possible to make – to change the digest of serial number (2 bytes) in the generated unlock code. For this purpose, pass in expanded options 80_PA (an icon «a wrench and a hammer»). The button is selected in the bright green color:



The auxiliary window of «*[80_PA] Advanced*» opens. In «Unlock code service structure» group we change «UC.Serial number stamp» value on other than the previous. It is recommended to use the «rnd» button for generation of accidental value. It is also possible to assign a tick for *Incremental random s/n stamps*, for generation of new value individually for each unlock code during the current session. We claim and save new value by means of «Accept» button



We repeat generation of unlock code with the new digest of serial number. The generated unlock code will be letters/digits from old.

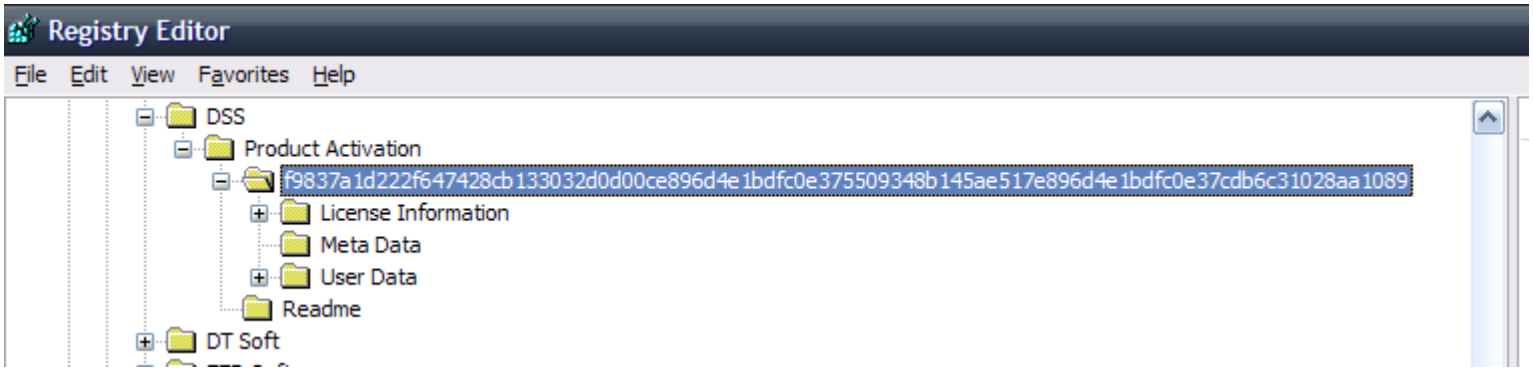
II. **Official revocation of license (revoke) as a method to drop a black list.** Using key /revoke for SecuROM version 8 and late 7 versions it is possible to achieve cleaning of "black list" of digests of serial numbers. For the first versions it is necessary to download the special revoke program.

III. **Informal (direct) deleting the license SecuROM PA for SecuROM ver.8 (for advanced users).** For this purpose, it is necessary to use the editor of the register of Windows (for example, standard regedit) and to know unique appid for each game (it can be pulled out by means of a debugger). Here, for each game in the register there will be «black list» of digests of serial number.

We come. The destination – a branch of activation of SecuROM PA: HKEY_CURRENT_USER\Software\DSS\Product Activation\

In this case, we are in a game branch «*Epic Mickey 2: The Power of Two*», for which is equal to HWID

f9837a1d222f647428cb133032d0d00ce896d4e1bdfc0e375509348b145ae517e896d4e1bdfc0e37cdb6c31028aa1089



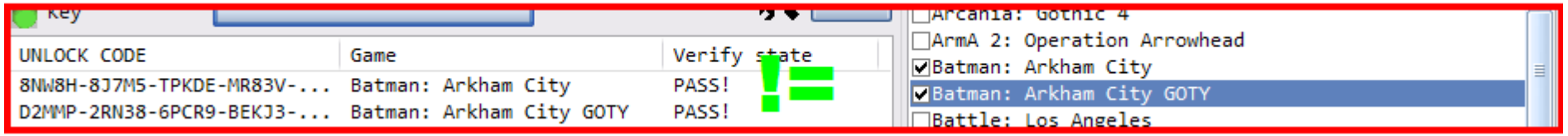
We delete the specified branch, thereby dropping the license. If you delete a root branch HKEY_CURRENT_USER\Software\, licenses for all games will be dropped.

15. We repeat registration procedure *(in the presence of errors!)*.

Special notes on the following games

1. **«Ys Foliage Ocean in Celceta». For a platoon of "Manual activation" replace in the folder with game dynamic library paul.dll and add lang.ini from archive** *80_PA___ALTERNATIVE_PAUL__for_ysc (ver 2.0.1.3).7z*

2. **GOTY (Game Of The Year) edition. Pay attention that the same games can differ on executions and respectively have different key kits.**



3. **(Trial mode) «The Travels of Marco Polo», «Sir Pudding Wiggly». For a platoon of "Manual activation" replace in the folder with game dynamic library paul.dll and add lang.ini from archive** *80_PA___ALTERNATIVE_PAUL__for_defeat_TRIAL_MODE and EA (ver 1.0.1.3).7z*

4. **(EA Game Authorization Management) «Command & Conquer: Red Alert 3», «Mass Effect», «Spore», «The Godfather II», «Mirror's Edge», «Mercenaries 2: World in Flames», «Burnout Paradise: The Ultimate Box», «Sims 3». For a platoon of "Manual activation" replace in the folder with game dynamic library paul.dll and add lang.ini from archive** *80_PA___ALTERNATIVE_PAUL__for_defeat_TRIAL_MODE and EA (ver 1.0.1.3).7z*



5. (Regional splitting) «TRON: Evolution» и «TRON: Evolution (RUSSIAN)». Pay attention that the same games can differ also on regional signs and respectively have different key sets (similarly, as in a case with GOTY edition).

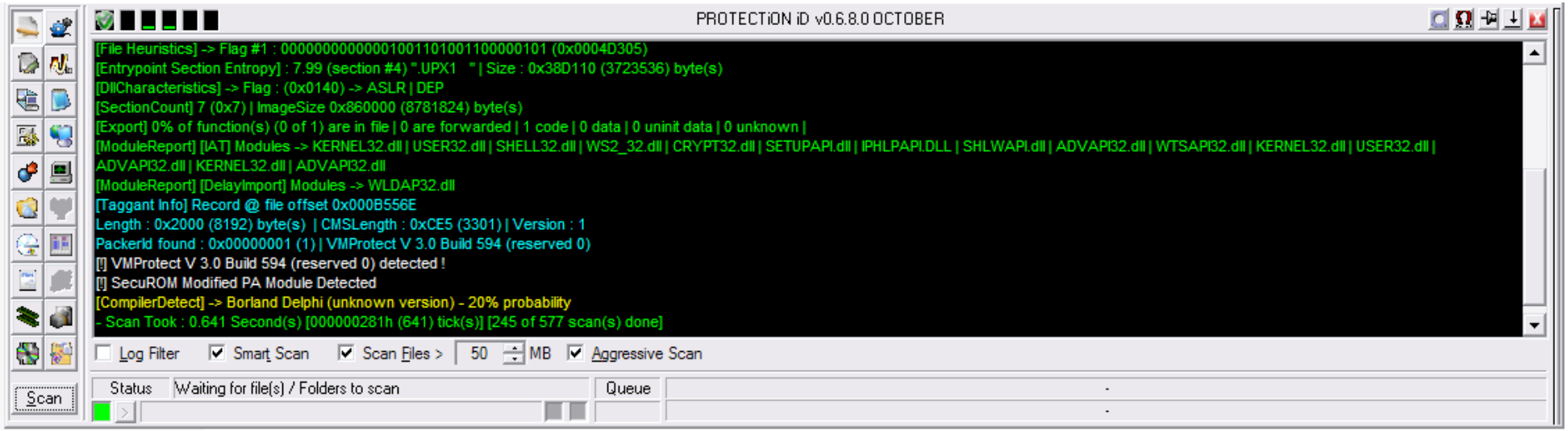
6. (Special situations) «GTA IV». Some users specified that in case of successfully complete activation game requires to insert a license compact disk into the drive. At the moment, when cracking SecuROM we didn't watch a similar situation. Usually in protection the condition OR, i.e. either online activation, or a compact disk is set. However, you can study the document *Sony DADC SecuROM vulnerability.pdf* for use of vulnerability in the module of check of compact disks SecuROM. «GTA IV EFLC». For EFLC there is no difference in case of start of executable files LaunchEFLC.exe or SteamActivation.exe (Steam) – in both cases key sets are identical.

7. (The version 08.13.xx - the latest, but little-known games of 2016 year) «Tale of Wuxia: Prequel» (taking into account updating from 17/10/16). For a platoon of «Manual activation» replace in the folder with a game dynamic library paul.dll and add lang.ini from archive *80_PA___ALTERNATIVE_PAUL__for_ysc (ver 2.0.1.3).7z*. It should be noted that for this SecuROM PA version the code of the procedure of activation is partially changed - it is recommended to remove/rename the active.exe and deactive.exe files, being in the folder with a game. In particular special changes have concerned RSA algorithm:

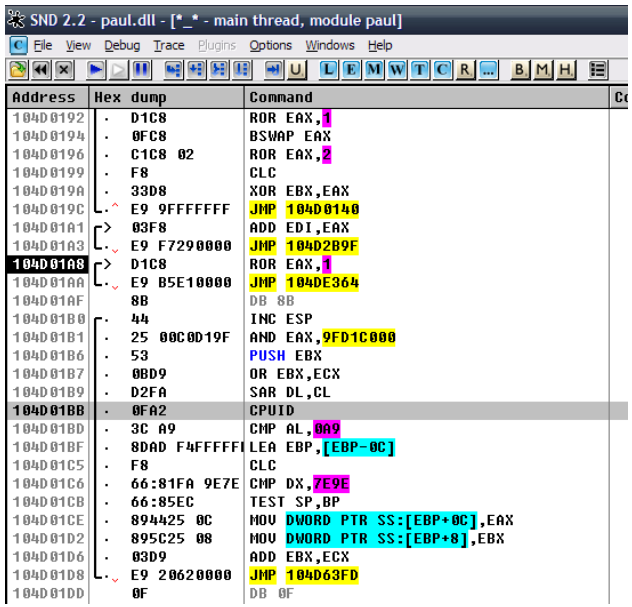
7.1 the module *n* (argument No. 4) is transferred in the distorted look now. Directly correct module is formed in the RSA procedure. It is supposed as: y = x^e mod (n*const)

7.2 an entrance ciphertext *x* (argument No. 2) and opened an exhibitor of *e* (argument No. 3) have traded places among themselves. During too time, *e* opened an exhibitor more undertakes not from an argument No. 2, and is directly in a body of an algorithm of RSA (inline).

7.3 paul.dll delivered with this game occupies 3,733,568 bytes (nearly 4 megabytes). There are bases to claim that paul.dll in this version is packed by DENUVO x86 (VMProtect 3.0 + SecuROM). Results of scanning by the utility of ProtectionIDv0.6.8.0 (OCTOBER,2016):

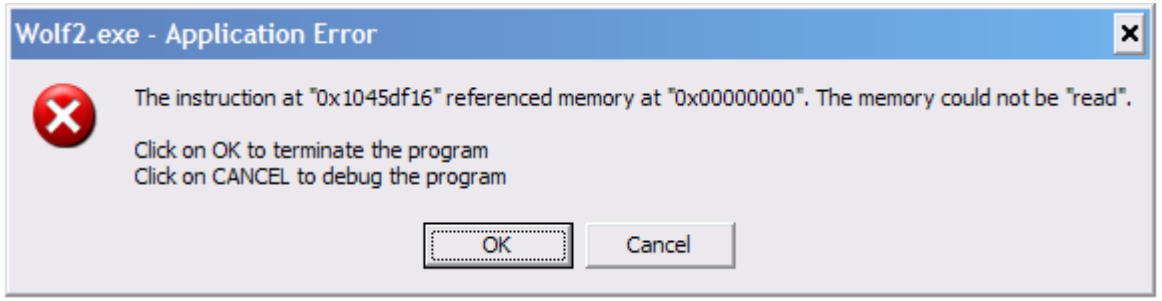


When debugging this paul.dll version instructions of CPUID often meet, and also the characteristic obfuscation is visible:

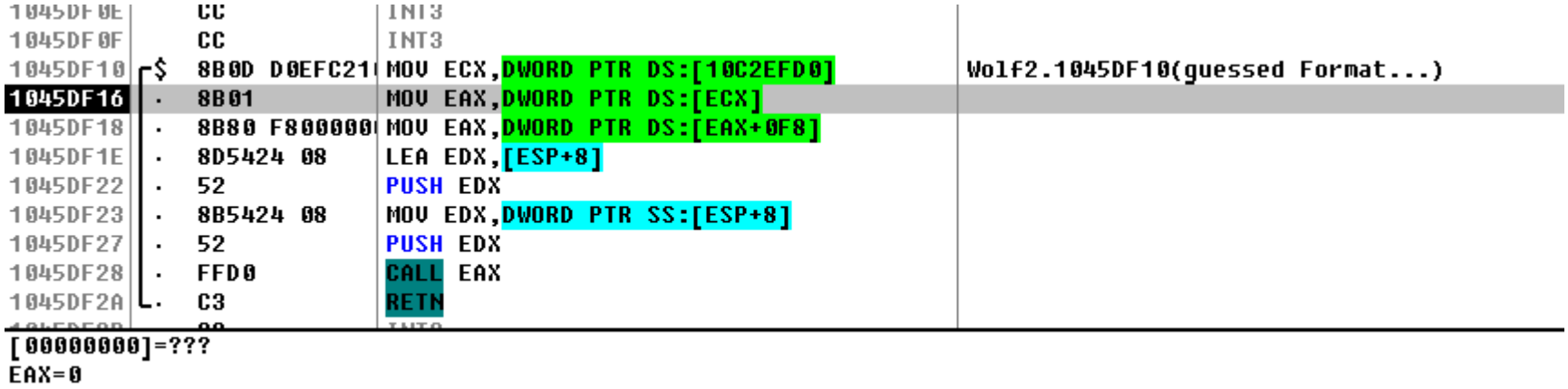


8. (The games which are officially not using online activation. By default, check of a license compact disk) «Dead Space», «Need for Speed: ProStreet», «Command & Conquer 3: Tiberium Wars», «Pro Evolution Soccer 2014», «Brave: The Video Game», «Lego Pirates of the Caribbean: The Video Game», «Operation Flashpoint: Red River». Despite the absence of the file-wrapper paul.dll in the folder with the specified games and appearance of a dialog of check of a compact disk in case of start, exists unobvious option of use 80_PA as an alternative checks of a license compact disk. All necessary cryptography sets are sewn already up in a game. Presumably under this vulnerability all games with versions of protection >= 7.3x get (probably to eat communication between developed, in the same time the virtual machine and technology of online activation SecuROM PA). For a platoon of «Manual activation» copy in the folder with a game dynamic library paul.dll and lang.ini from archive *80_PA___ALTERNATIVE_PAUL__for_special_variants(GTA4_ver 1.0.1.14).7z* (use of archive *80_PA___ALTERNATIVE_PAUL__for_defeat_TRIAL_MODE and EA (ver 1.0.1.3).7z* is in certain cases allowed).

9. 9. (The games which are officially not using online activation. By default, check of a license compact disk is cocked. Special case!). «Wolfenstein (2009)» (Version of a game: 0.91.25.7022). For a platoon of «Manual activation» copy in the folder with a game dynamic library paul.dll and lang.ini from archive *80_PA___ALTERNATIVE_PAUL__for_special_variants(GTA4_ver 1.0.1.14).7z*. Activate a game with the help 80_PA. If after activation Windows the message of «Application error» readings, signaling on an error when reading target address:



(In a debugger it is possible to watch it in the place provided below)



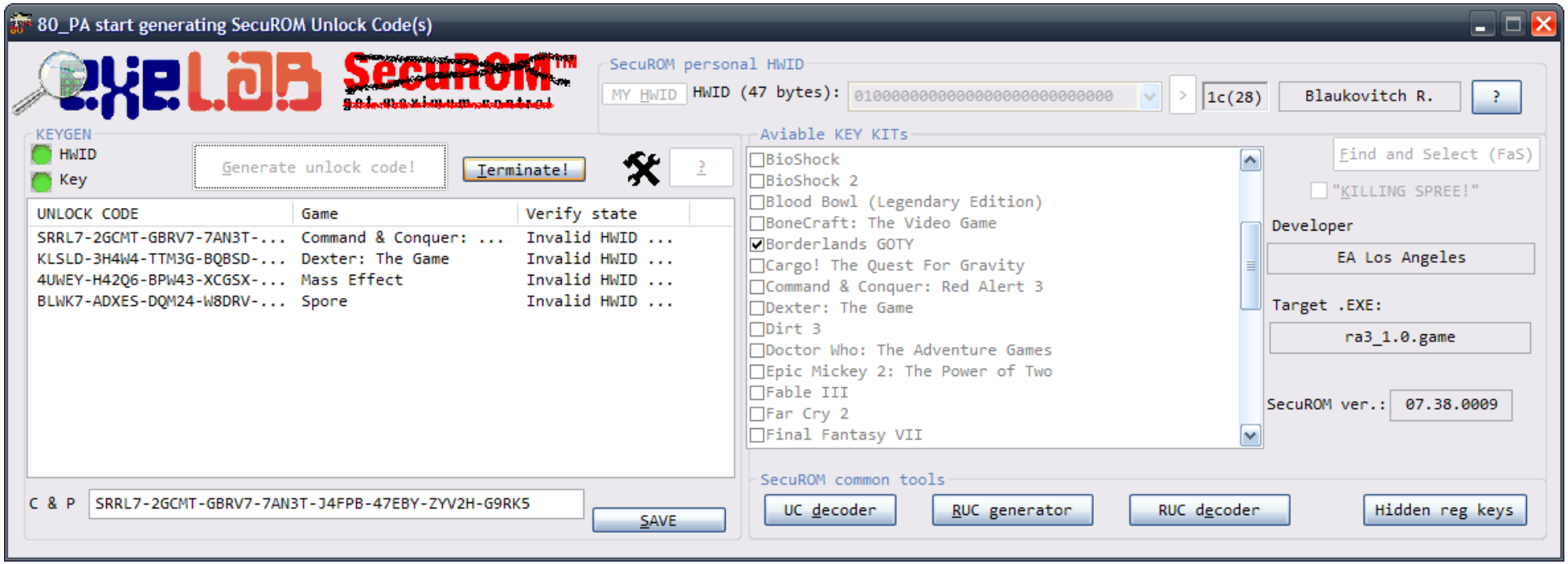
Decision: close MessageBox and rename (or delete) paul.dll in the folder with a game. Launch Wolf2.exe again – the error shall disappear. Check of a license compact disk in case of correct online activation won't be active.

10. (Games from «Telltale Games»). Use the built-in implementation of API from paul.dll. Ignore the external file of the specified library. Interface elements in case of activation are drawn by means of the built-in browser.

11. http://joyoland.com/, 北京欢乐百世科技有限公司, Nightshade (百花百狼), Norn9 (命運九重奏), Empire of Angels IV (天使帝國四), The Legend of Heroes: Trails from Zero (《零之轨迹》), The Legend of Heroes: Trails to Azure (英雄伝説 碧の軌跡：改), YS7 (イソ7), YSC (イース セルセタの樹海) Replace PAUL.DLL (DENUVO Gmbg) and lang.ini with earlier versions from the /80_PA addons folder included with the 80_PA SecuROM keygen. These are the latest versions of SecuROM 08.13.076 (2018).

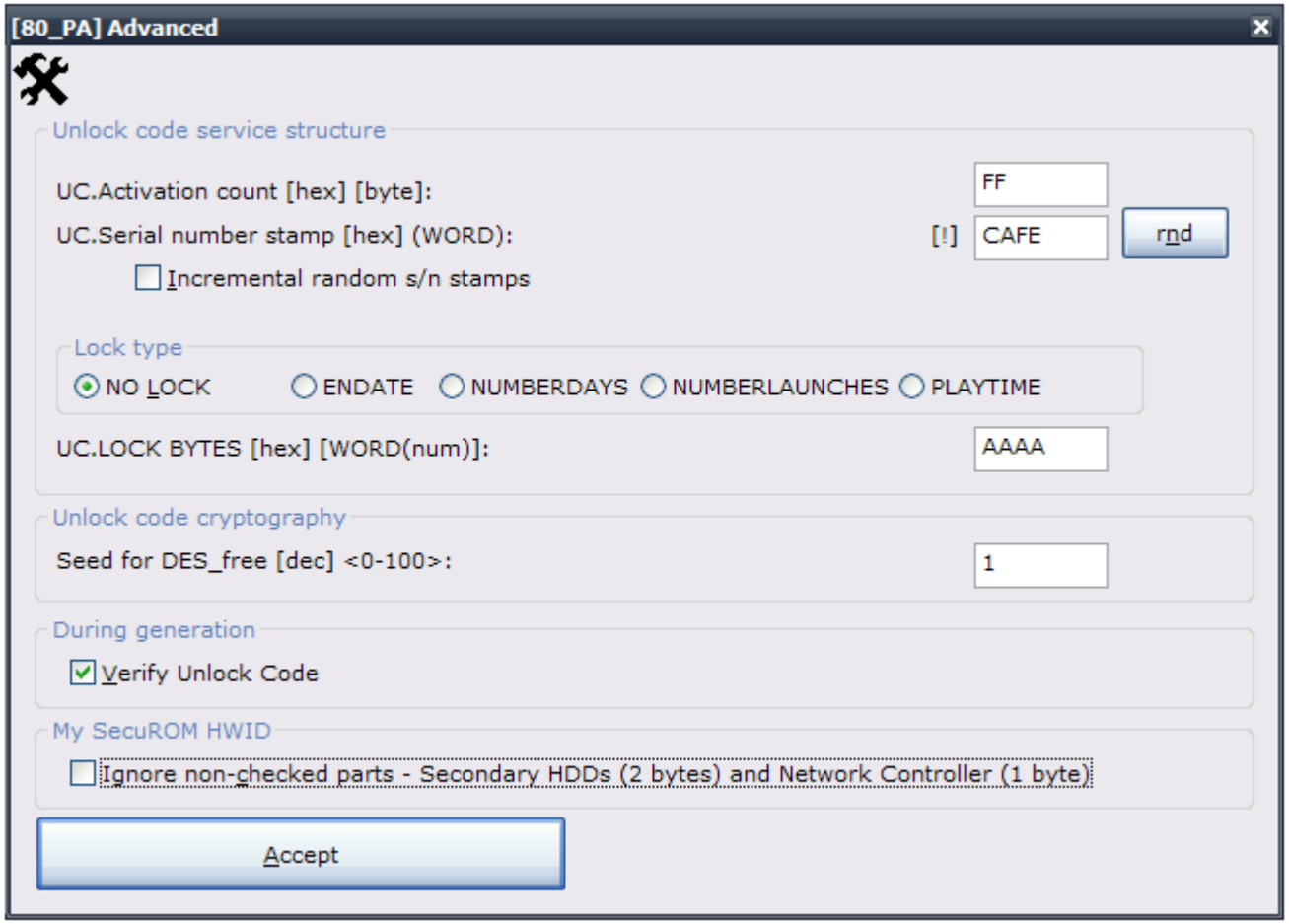
Review of windows.

Secondary window of the program 80_PA. There is a generation of unlock code for the selected games. The active HWID is displayed. Access to remaining windows is provided.



The options directly influencing finite unlock code are shown in a window [80_PA]Advanced

It is recommended to change value only in the edit box Serial number stamp



UC.Activation count – is probable, the number of activations on your HWID in the database of the server of Sony of DADC AG. This data only for information.

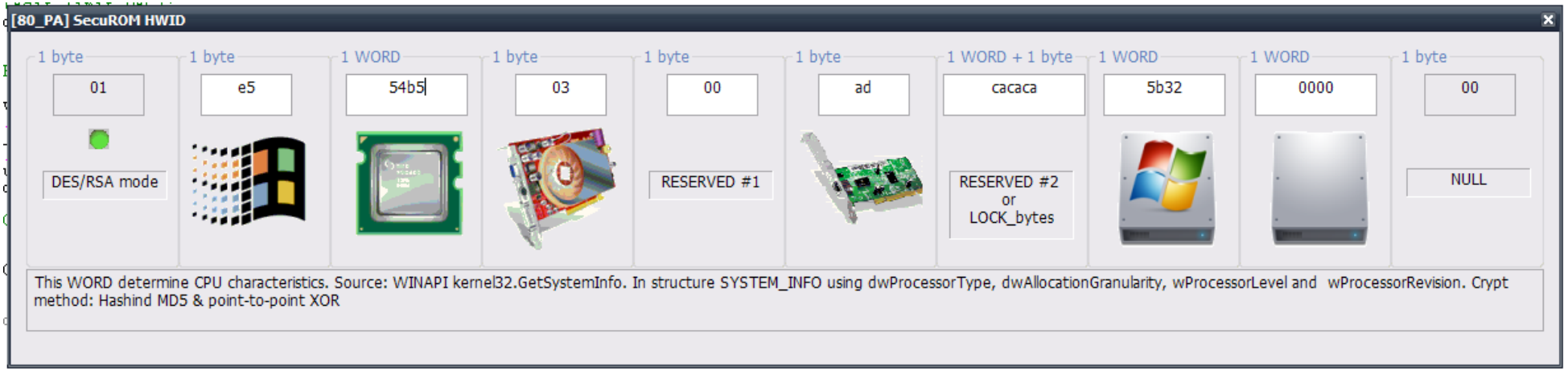
Lock type – lock type with a binding by quantity in **UC.LOCK BYTES.** Always you cock this parameter as **NO LOCK**, otherwise unlock code will impose restrictions for start.

Seed for DES_free – random generated seed for primary (free) key of DES. Value of seed influences unless search speed in a cycle in case of decryption of unlock code (the more value – the one iteration more).

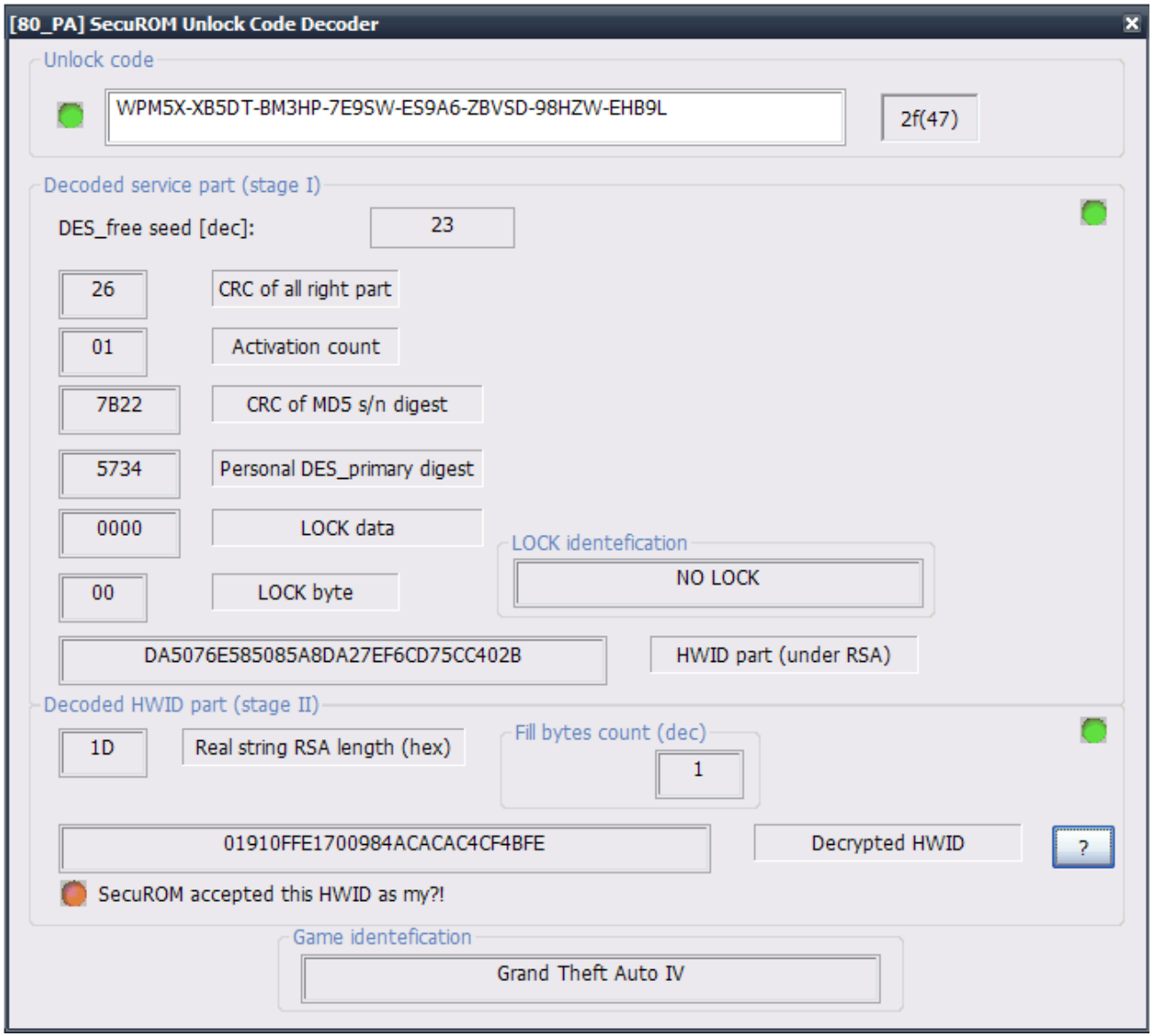
Ignore non-checked parts - Secondary HDDs (2 bytes)* and Network Controller (1 byte) – compulsorily not to calculate hashes of the SecuROM HWID components which influence the end result of activation. In this case it is about pa_raw_hwid. Network_nfo_hashik and pa_raw_hwid. Secondary_HardDisks_serial_nfo_hashik. In all probed games, SecuROM doesn't read change of these values as violation of license online activation. Let's say in case of pa_raw_hwid. Network_nfo_hashik hash (data of the network interface card) change can happen in case of normal switching on or switching off of a network by the user, i.e. the fact of change of the most network interface card is absent. This nuance obviously was also considered when checking all SecuROM HWID. It is recommended to use this option if you are sure that activation flies because of incorrect HWID values. After its application update the HWID by clicking of the «MY HWID» button in a primary window.

*** WORD или 2 bytes**

Detail layout of the active HWID is displayed in a window [80_PA]SecuROM HWID (data can only be looked, changes in this window don't remain). Correctly generated unlock code will be bound only to your computer and nobody else won't be able to use it (except for emulation of HWID values, using, for example, WinAPI hook)

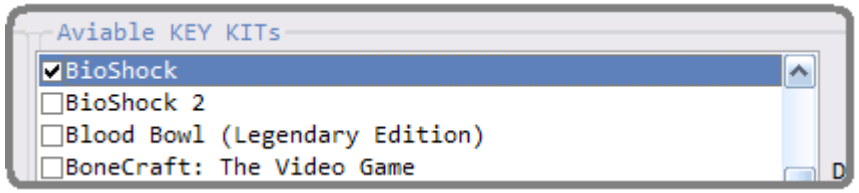


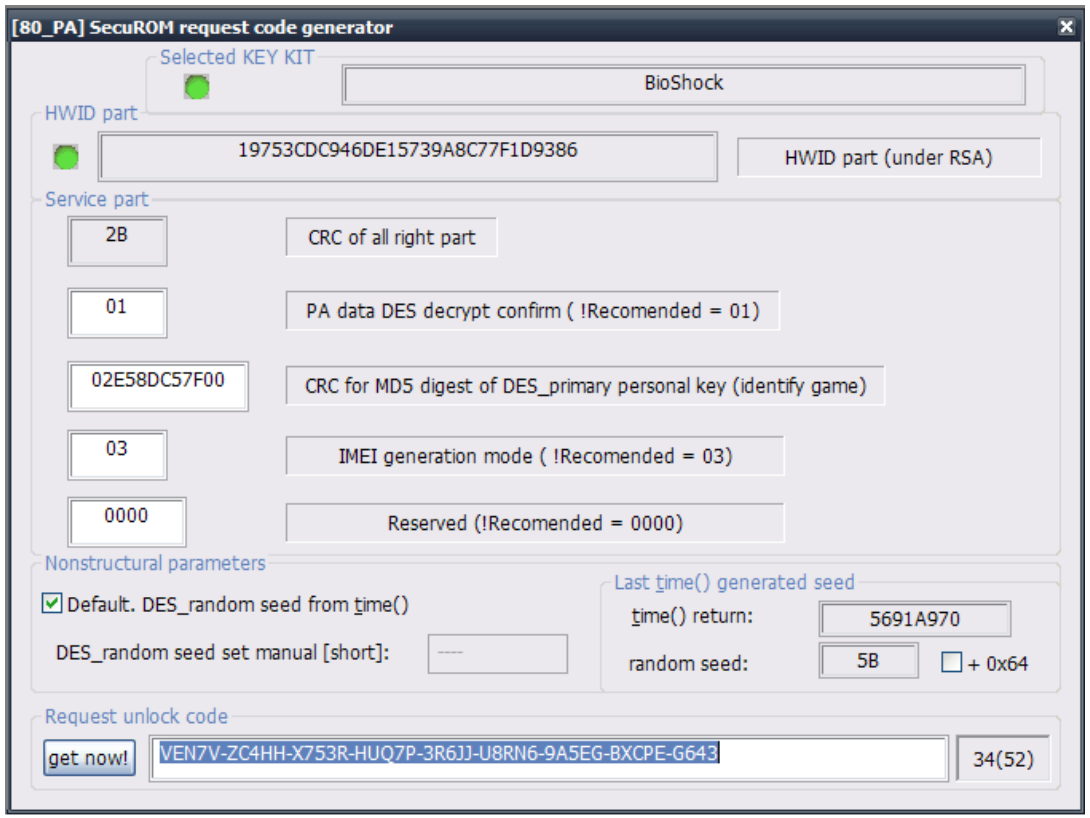
Fast decryption of any unlock code is carried out in a window [80_PA] SecuROM Unlock Code Decoder



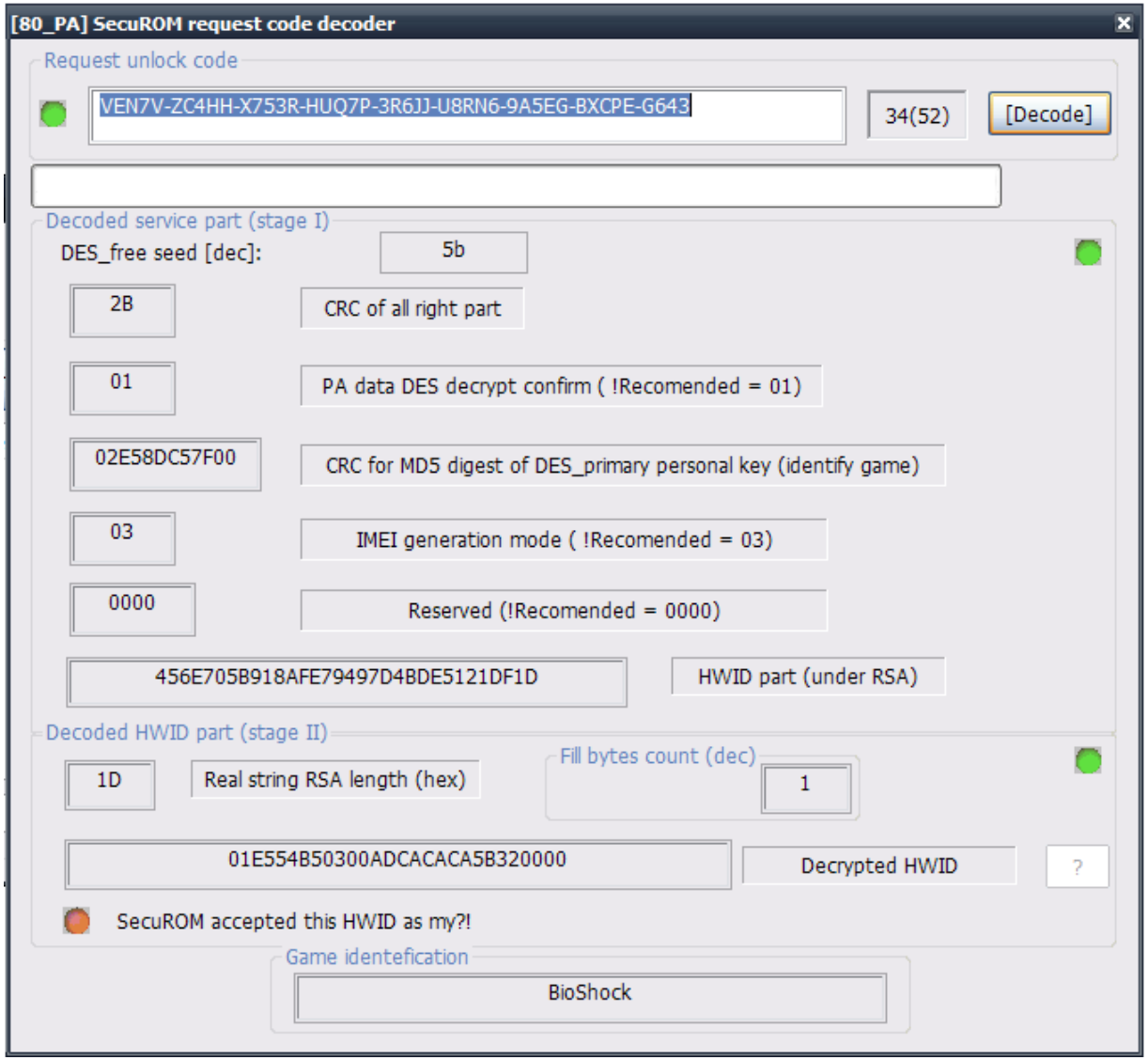
80_PA EN (machine translation)

To use the unlock requestcode generator (code request) [80_PA] SecuROM request code generator need to select game from a primary window beforehand.

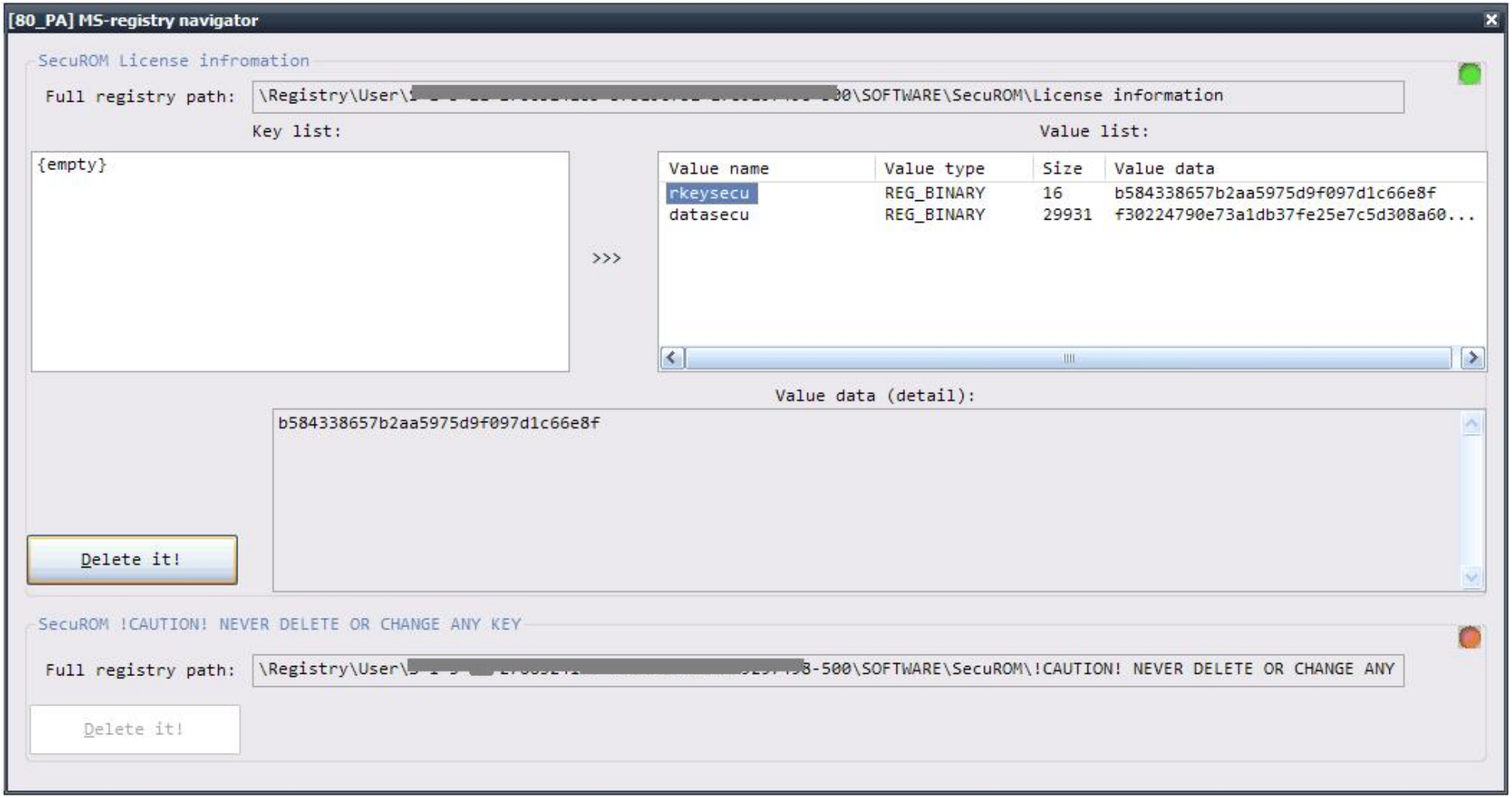




It is possible to decode any unlock request code (as it does the server of activation of Sony DADC AG) in a window [80_PA] SecuROM request code decoder.

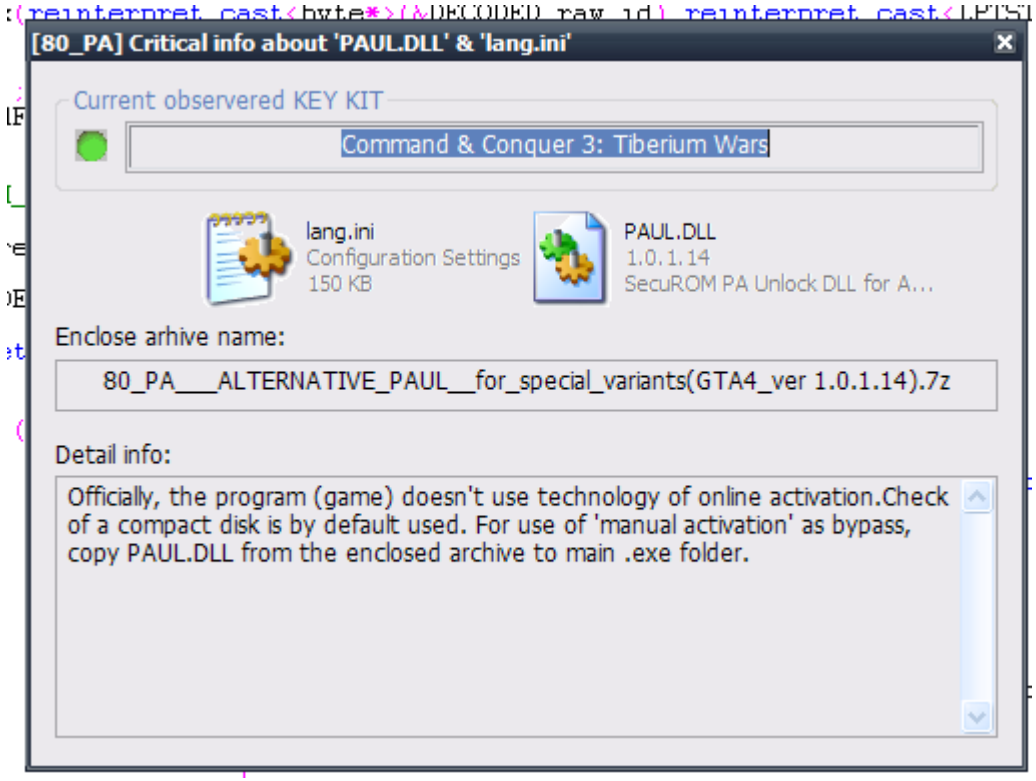


Pulling of License information this from the hidden key, and also its deleting together with !CAUTION! NEVER DELETE OR CHANGE ANY KEY is carried out in a window [80_PA] by MS-registry navigator.

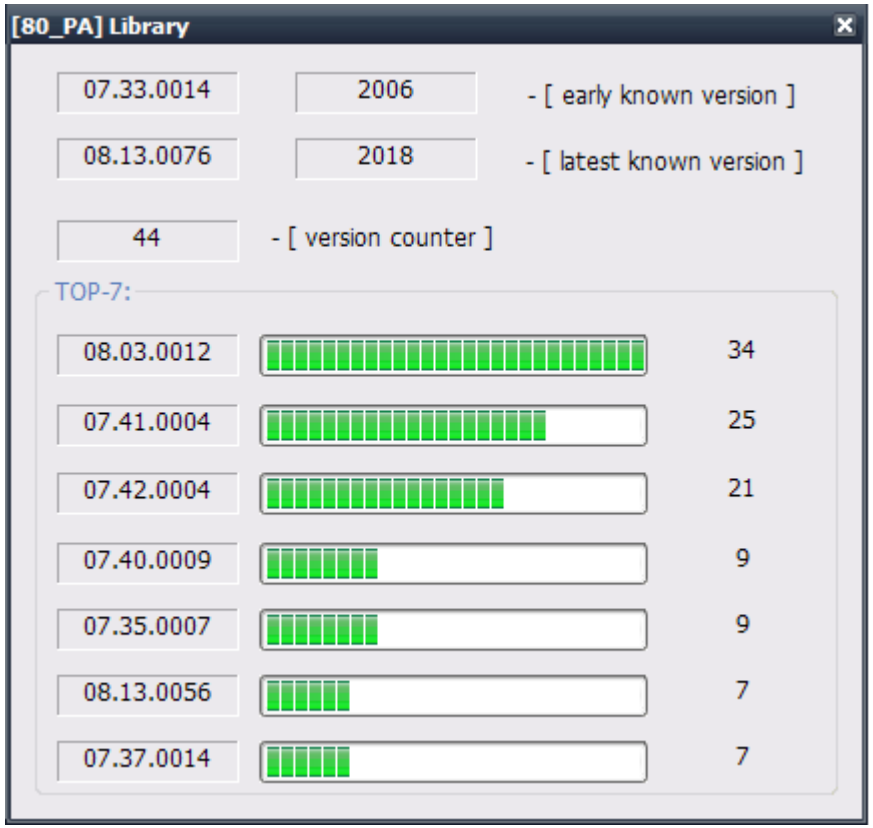


The window [80_PA] Critical info about 'PAUL.DLL' & 'lang.ini' is directly connected to icons 

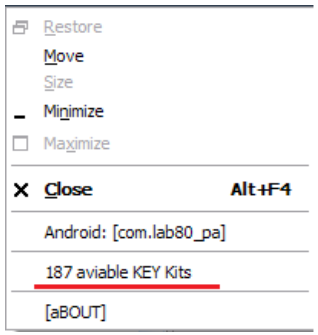
, which are displayed in a primary window. Actually the window contains the picture with the «lang.ini» and «paul.dll» files, and also informs on the recommended enclosed archive from the «*80_PA addons*» folder (if in your case problems are watched, try to pick up archive with other versions of library-wrapper «paul.dll») from which it is necessary to get the stipulated files for a platoon of "Manual Activation". Also, the important additional information necessary for incorrect completion of the procedure of activation can be specified.



In later 2.0 versions, statistics from the [80_PA] Library for known versions of Sony DADC AG SecuROM appeared.



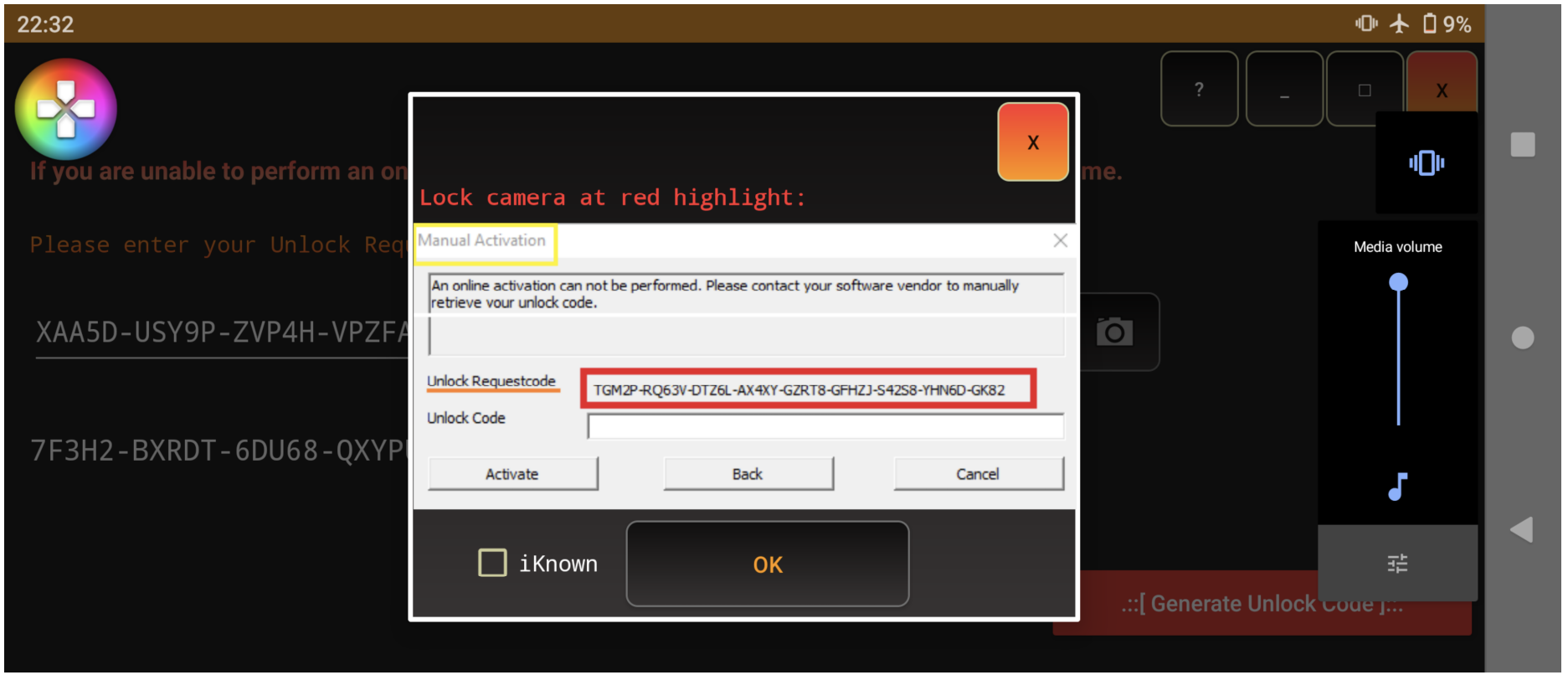
Available through the 80_PA SecuROM keygen main menu:



80_PA EN (machine translation)

Android packet: com.lab 80pa

It was written much later, part of keygen code was rewrite for the correct compilation in LLVM. Further, the changes were transferred to the V.2.0 version to Windows (Intel C/C ++). No different from V.2.0 with the exception of the method of entering the "REQUEST Unlock Code" - through the smartphone camera by text recognizing (aim 1G46).



https://www.amazon.com/gp/product/B0DQZWXQQ8 https://www.rustore.ru/catalog/app/com.lab80_pa

Android 4.0+ (Ice Cream Sandwich, API 14) or higher

MacOSX «Cider» / Linux Wine

Cider is an application in Mac OS X that runs Windows games covered by SecuROM 7-8 in this environment. The behavior of 80_PA, in this case, is SIMILAR to the Windows environment, in which your personal HWID is taken and an unlock code is generated. However, in case of problems with generating an unlock code or running 80_PA in Linux/Mac OS X via Wine/Cider emulators (at the moment, there are no known native games/programs for Linux/Mac OS X that have ever used SecuROM DRM), it is easier to run 80_PA SecuROM keygen on a regular computer with Windows XP-11 installed and generate a response unlock code for the known unlock code request, which is specified in the activation window manually (manual activation).

For early versions of 80_PA in Linux, you needed to install mfc42.dll:

```
sudo add-apt-repository ppa:ubuntu-wine/ppa
sudo apt-get update && sudo apt-get upgrade
winetricks mfc42
```

In later versions, this option is no longer needed.

Disclosure of an initial code of generation of HWID and structures of unlock (request) code.

```
#include "80_PA.h"
/*
                                                ELF present:

       ---------      ---------              ------------         -----------
      ----     ----  ----   ----             ----      ----      ----     ----
      ----     ----  ----   ----             ----      ----     ----       ----
      ----     ----  ----   ----             ----     ----      ----       ----
       ----------   ----   ----             ------------      ----       ----
      ----     ----  ----   ----             ----                -----------
      ----     ----  ----   ----             ----               ----       ----
      ----     ----  ----   ----             ----               ----       ----
        ---------     ---------   ----------- ----              ----       ----
                  Russian Hackers SecuROM PA (online-activation) project
                                                EXELAB.RU
*/
// LOCK идентификация
#define PA_UNLOCK_CODE_lock__NO_LOCK 0 //НЕТ БЛОКИРОВКИ
#define PA_UNLOCK_CODE_lock__LOCK_ENDDATE 4 //БЛОКИРОВКА ПО КОНЕЧНОЙ ДАТЕ ПОЛЬЗОВАНИЯ
#define PA_UNLOCK_CODE_lock__LOCK_NUMBERDAYS 3 //БЛОКИРОВКА ПО ДНЯМ
#define PA_UNLOCK_CODE_lock__LOCK_NUMBERLAUNCHES 2 //БЛОКИРОВКА ПО КОЛИЧЕСТВУ ЗАПУСКОВ
#define PA_UNLOCK_CODE_lock__LOCK_PLAYTIME 1 //БЛОКИРОВКА ПО ВРЕМЕНИ В ИГРЕ
```

```
// структура SecuROM unlock code //
#pragma pack(1)
typedef      struct sc_lock_part //LOCK - опции блокировки ключа
             {
             unsigned short T80_LOCK_INT_DATA; // данные блокировки (2 байта)

             byte T80_LOCK_TYPE_IDENT; //индетификатор типа блокировки (1 байт)
             }lock_part;

typedef      struct sc_imei_part //зашифрованный HWID
      {
             byte T80_IMEI[15]; //зашифрованное значение HWID(15 байт)
             byte T80_IMEI_as_RSA_string_Length; //длина шифрованного значения HWID в строчном ASCII-формате (1 байт)
      }imei_part;

 typedef struct ELF_80_PA_UNLOCK_CODE
{
byte T80_CRC_of_right_part; //CRC правой части (1 байт)
byte T80_Activate_count; //Контрольный байт активации (1 байт)
unsigned short T80_CRC_of_MD5_Serial_num; //Дайджест серийного номера (2 байта)
byte T80_CRC_of_MD5_DES_PRIMARY_key_digest[2]; // Дайджест от appid ( 2 байта)

lock_part lock;
imei_part imei;

}T_80_unlock;

//структура unlock requestcode //
typedef struct ELF_80_PA_REQUEST_UNLOCK_CODE
{
byte T80_PA_CRC_Polynom; // CRC правой части (1 байт);
byte T80_PA_DES_Success_decrypt_confirm; //DES success (1 байт)
byte T80_PA_CRC_MD5_digest_of_DES_prep[6]; // Дайджест от appid (6 байт)
byte T80_PA_REQUEST_MODE_generation; //режим генерации (1 байт)
byte T80_PA_reserved_unkwnown[2]; //неизвестно. [возможно переходящие LOCK BYTE] (2 байта)
imei_part imei;
}T_80_request_unlock, *pT_80_request_unlock;
```

80_PA EN (machine translation)

```
// структура SecuROM HWID //
typedef struct RAW_MACHINE_ID
{
	bool IsRealTimeGenerated;
	byte Version_nfo_hashik;
	WORD System_nfo_hashik;
	byte VideoBoard_nfo_hashik;
	byte Reserved1;
	byte Network_nfo_hashik;
	WORD Reserved2;
	byte Reserved21;
	WORD System_HardDisk_serial_nfo_hashik;
	WORD Secondary_HardDisks_serial_nfo_hashik;
	byte Null_terminant;
}pa_raw_hwid, *ppa_raw_hwid;
#pragma pack()

// Маска проверки объектов HWID //
typedef struct VERIFY_MASK_HWID
{
	bool Verify_IsRTG_flag;
	bool Verify_Version_nfo;
	bool Verify_System_nfo;
	bool Verify_VideoBoard_nfo;
	bool Verify_Reserved1;
	bool Verify_Network_nfo;
	bool Verify_Reserved2;
	bool Verify_HardDisk_serial_nfo;
	bool Verify_HardDisk_secondary;
}PA_verify_mask_hwid;
PA_verify_mask_hwid pa_current_config = {1,1,1,0,0,0,1,0}; // Дефолтное состояние проверки HWID, зашитое в SecuROM
// Процедура сборки SecuROM HWID //
void Get_raw_machine_ID(ppa_raw_hwid raw_ID)
{
	/* 1 step */
	OSVERSIONINFO osinfo;
```

```
SYSTEM_INFO sysinfo;
D3DADAPTER_IDENTIFIER9 gpu_info;

PIP_ADAPTER_INFO pAdapterInfo;
ULONG ulOutBufLen = (sizeof (IP_ADAPTER_INFO )*8);



unsigned long MD5_Data[32]; //128 (0x80) bytes !!!

memset((void*)raw_ID,0,sizeof(RAW_MACHINE_ID));

raw_ID->IsRealTimeGenerated=true;

 /* 1 step */ //(информация об ОСи)
memset(&osinfo,0,sizeof(OSVERSIONINFO));
memset(&MD5_Data[0],0,sizeof(MD5_Data));
MD5_CTX md5context;

osinfo.dwOSVersionInfoSize = sizeof(OSVERSIONINFOEX);
::GetVersionEx(&osinfo);


                    MD5_Init(&md5context);


                    md5context.Nl = MD5_Cont_Size;
                    md5context.Nh = 0;
                    md5context.num=MD5_DIGEST_LENGTH;
                    md5context.data[0] = osinfo.dwMajorVersion;
                    md5context.data[1] = osinfo.dwMinorVersion;
                    md5context.data[2] = osinfo.dwBuildNumber;
                    md5context.data[3] = osinfo.dwPlatformId;

                    MD5_Final((unsigned char*)&MD5_Data[0], &md5context);


               QUCK_XOR_RAW_DATA(&raw_ID->Version_nfo_hashik,(byte*)&MD5_Data[0],sizeof(test_raw_hwid.Version_nfo_hashik));

/* 2 step */ //(информация об установленном процессоре)
::GetSystemInfo(&sysinfo);
```

```
                    MD5_Init(&md5context);

                    md5context.Nl = MD5_Cont_Size;
                    //md5context.Nh = 0;
                    md5context.num=MD5_DIGEST_LENGTH;
                    md5context.data[0]=sysinfo.dwProcessorType;
                    md5context.data[1]=sysinfo.dwAllocationGranularity;
                    md5context.data[2]=sysinfo.wProcessorLevel;
                    md5context.data[3]=sysinfo.wProcessorRevision;

                    MD5_Final((unsigned char*)&MD5_Data[0], &md5context);

                QUCK_XOR_RAW_DATA((byte*)&raw_ID->System_nfo_hashik,(byte*)&MD5_Data[0],sizeof(test_raw_hwid.System_nfo_hashik));

    /* 3 step */ //(информация об установленной видеокарте)
    HMODULE h_lib = LoadLibrary("d3d9.dll");

        if (h_lib != NULL)
        {
        D3D9Create=(d3d9_create)GetProcAddress((HMODULE)h_lib,"Direct3DCreate9");

        PDIRECT3D9 d3d9struct = D3D9Create(D3D_SDK_VERSION);

        d3d9struct->TABLE_d3d9->GetAdapterIdentifier(d3d9struct, D3DADAPTER_DEFAULT,D3DENUM_WHQL_LEVEL, &gpu_info);

        FreeLibrary(h_lib);

                    MD5_Init(&md5context);

                    md5context.Nl = MD5_Cont_Size;
                    //md5context.Nh = 0;
                    md5context.num=MD5_DIGEST_LENGTH;
                    md5context.data[0]=gpu_info.VendorId;
                    md5context.data[1]=gpu_info.DeviceId;
                    md5context.data[2]=gpu_info.SubSysId;
                    md5context.data[3]=gpu_info.Revision;

                    MD5_Final((unsigned char*)&MD5_Data[0], &md5context);

        QUCK_XOR_RAW_DATA(&raw_ID->VideoBoard_nfo_hashik,(byte*)&MD5_Data[0],sizeof(test_raw_hwid.VideoBoard_nfo_hashik));

        }
```

```
/* 4 step */ //(информация о сетевой карте) offline/online mode
h_lib = LoadLibrary("IPHLPAPI.dll");
      if (h_lib != NULL)
      {

             pAdapterInfo = (IP_ADAPTER_INFO *) malloc(sizeof (IP_ADAPTER_INFO)*8);
            IPGetAdaptersInfo=(IPHLPAPI_GetAdaptersInfo)GetProcAddress((HMODULE)h_lib,"GetAdaptersInfo");




      IPGetAdaptersInfo(pAdapterInfo, &ulOutBufLen);

            FreeLibrary(h_lib);

            MD5_Init(&md5context);

            md5context.data[1]=0;
            memcpy(&md5context.data[0],pAdapterInfo->Address,sizeof(pAdapterInfo->Address));
            md5context.Nl = MD5_Cont_Size_for_IPHLAPI;
            md5context.num=MD5_DIGEST_LENGTH-10;


            md5context.data[2]=0;
            md5context.data[3]=0;


                  MD5_Final((unsigned char*)&MD5_Data[0], &md5context);

            QUCK_XOR_RAW_DATA(&raw_ID->Network_nfo_hashik,(byte*)&MD5_Data[0],sizeof(test_raw_hwid.Network_nfo_hashik));

      free((void*)pAdapterInfo);
      }
/* 5 step*/ //в оригинале там цикл опроса через GetDriveType(c:..z:) с первым попавшимся HDD (читай ищется системный HDD)
            char NameBuffer[MAX_PATH];
            char SysNameBuffer[MAX_PATH];
            DWORD VSNumber=0;
            DWORD MCLength=0;
            DWORD FileSF=0;
            char disk[3];
            disk[1]=*":";
```

```
                disk[2]=*"\\";
                disk[3]=0x0u;


            for( disk[0] = *"c"; disk[0] <= *"z";disk[0]=(byte)disk[0]+1)
            {


                if (::GetDriveType(&disk[0]) == DRIVE_FIXED)
                {
::GetVolumeInformation(&disk[0],NameBuffer, sizeof(NameBuffer), &VSNumber,&MCLength,&FileSF,SysNameBuffer,sizeof(SysNameBuffer));
                    break;
                }

            }
            __asm //SWAP VSNumber
            {
                MOV EAX, DWORD PTR SS:[VSNumber]
                BSWAP EAX
                MOV DWORD PTR SS:[VSNumber], EAX
            }

                MD5_Init(&md5context);

                md5context.Nl = (MD5_Cont_Size/4);
                md5context.num=(MD5_DIGEST_LENGTH/4);
                md5context.data[0]=VSNumber;
                md5context.data[1]=0;
                md5context.data[2]=0;
                md5context.data[3]=0;


                MD5_Final((unsigned char*)&MD5_Data[0], &md5context);

QUCK_XOR_RAW_DATA((byte*)&raw_ID-
>System_HardDisk_serial_nfo_hashik,(byte*)&MD5_Data[0],sizeof(test_raw_hwid.System_HardDisk_serial_nfo_hashik));

                raw_ID->Null_terminant=NULL;


}
```

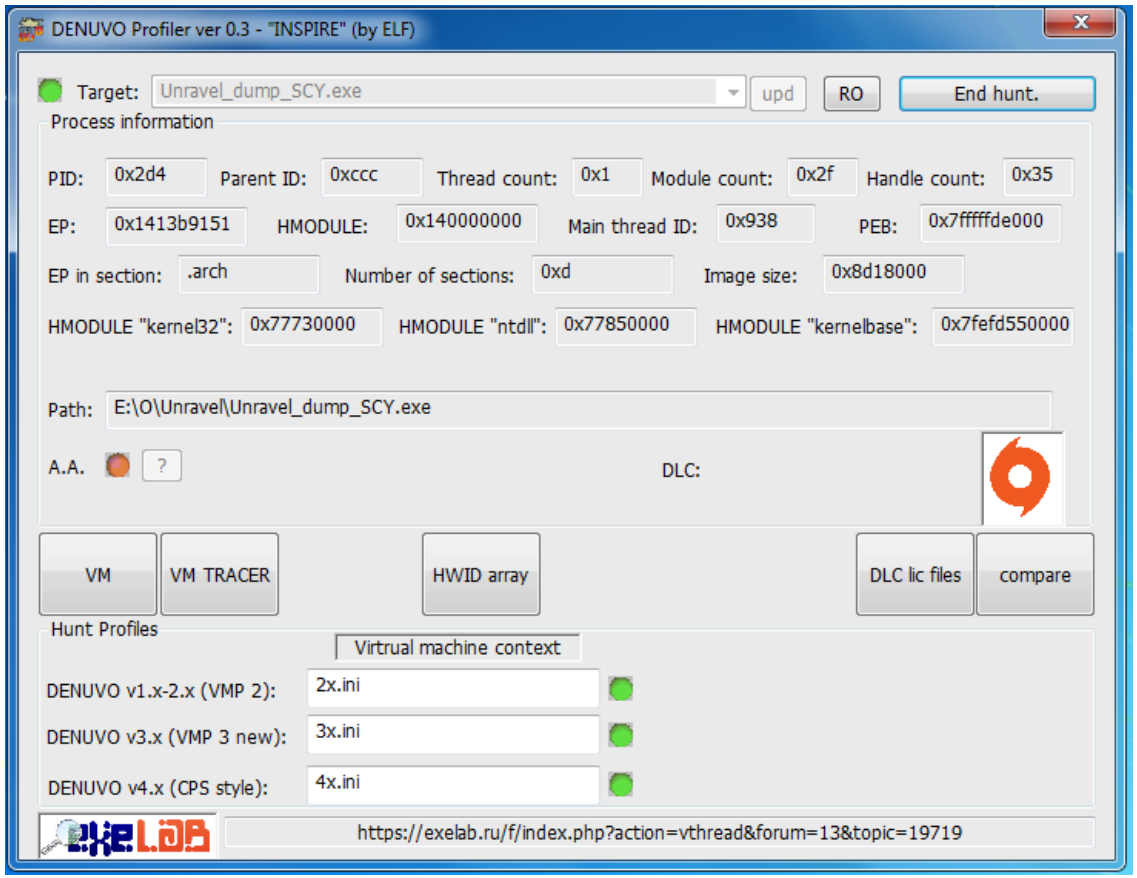
80_PA EN (machine translation)

```
// вспомогательная функция шифрования данных в HWID
void __inline __fastcall QUCK_XOR_RAW_DATA (byte* Desdonation, byte* MD5_CTX_start, DWORD Xoring_size)
{
	for(unsigned long i=1; i<=(sizeof(MD5_LONG)*4); i++)
	{
		for(unsigned long k=0; k<=Xoring_size-1; k++)
		{
			*(Desdonation+k) ^= *(MD5_CTX_start++);
		}

	}
}
```

Other projects.

IRL it is more than them



DENUVO_Profiler (DProfiler) **– be preparing to dominate to one of famous DRM on the world.**



diff_trace – the program for comparing of two trace-logs (trace) saved by OllyDbg 2.x debugger (or its analogs). *diff_trace* was used for cracking of the module of check of geometry of the disks SecuROM. Extends with source texts. (https://exelab.ru/f/index.php?action=vthread&forum=3&topic=20942)



SPR I (SecuROM Profiler) – the program assistant for operation with the virtual machines (virtual machine) SecuROM v7.3x – v8.x. Includes monitoring of anti-attach (A.A.) for association to already launched protected process. (https://exelab.ru/F/index.php?action=vthread&forum=13&topic=19719)


DUNE 2009 (DUNE_LAUNCH.exe) – the original game DUNE 2000 from WestWood Studios with the changed engine. (http://rutracker.org/forum/viewtopic.php?t=3637042)



Dark Colony – the original cured game Dark Colony (Alcohol of 120% for assembling of an image isn't required!). Some errors and bugs are corrected. The **DC_SAV** programs (the editor of saving) and **DC_RET** are added (for incorrect resetting to game when switching Alt+Tab) (http://rutracker.org/forum/viewtopic.php?t=3683906)


DOoM 2 game – the frivolous unpretentious game, written in 2006 ago on VB 6 on destruction of "Dom-2". (http://rutracker.org/forum/viewtopic.php?t=3703290)



HEIDENHAIN TNCremoNT (Plus) + TeleService – cracked versions of famous programs for data exchange with CNC machines TNCremoNT and TeleService.(https://rutracker.org/forum/viewtopic.php?t=5426612)

Atlassasin Jira & Confluence private crack – latest versions of Atlassian products + plugins (from Marketplace).

CIMCO Software – CIMCO A/S. CIMCO Edit 2025, 8, 6 + file transfer. **CTranslate** (CIMCO Translate) – unofficial view & edit CIMCO language files (you can add own CIMCO translation).

Google Chrome 141 for Windows 7 (WebGPU support)

https://habr.com/ru/articles/752692/

https://habr.com/ru/articles/789120/

https://github.com/Blaukovitch/GOOGLE_CHROME_Windows_7_CRACK

https://rutracker.org/forum/viewtopic.php?t=6384596

About the project 80_PA. Feedback.

Author of technology 80_PA, crack SecuROM: **ELF**

We express huge gratitude: random (manager of key kits)

Archer (solutions explorer),

 int (PA unlock page)

reversecode (DES)

Nightshade (advices)

mak (old SecuROM info)

Haoose (www.antistarforce.com)

painter *(v00doo)*

mysterio

gig3x , HardcoreHacker[HcH], Tihamér Szabó

Thanks to all remaining participants for support:

OnLyOnE, ARCHANGEL, Bronco, Vovan666, VodoleY, DimitarSerg, DenCoder, Gideon Vi, MasterSoft, BAHEK, ClockMan, SReg, [Nomad], daFix, 4kusNick, Ara, Smon, DillerInc, Dart Raiden, zeppe1in, kioresk, SER[G]ANT, DeZoMoR4iN, ajax, vovanre, SharkXXL, mysterio, too87264 *and all remaining whom I didn't list!*

Separate noble thanks: Sony DADC AG (*now* «Denuvo Software Solution GmbH») ☺

https://exelab.ru/f/PAunlock/

https://exelab.ru/f/index.php?action=vthread&forum=13&topic=19719

http://exelab.ru/rar/dl/CRACKLAB.rU_107.rar

https://youtu.be/AcVTF1HfTb8
https://youtu.be/x6M5bOvv0Fg

http://rutracker.org/forum/viewtopic.php?t=5116975

http://antistarforce.com/forum/8-16870-1

https://xakep.ru/2015/08/07/securom/
https://xakep.ru/2019/04/19/denuvo/

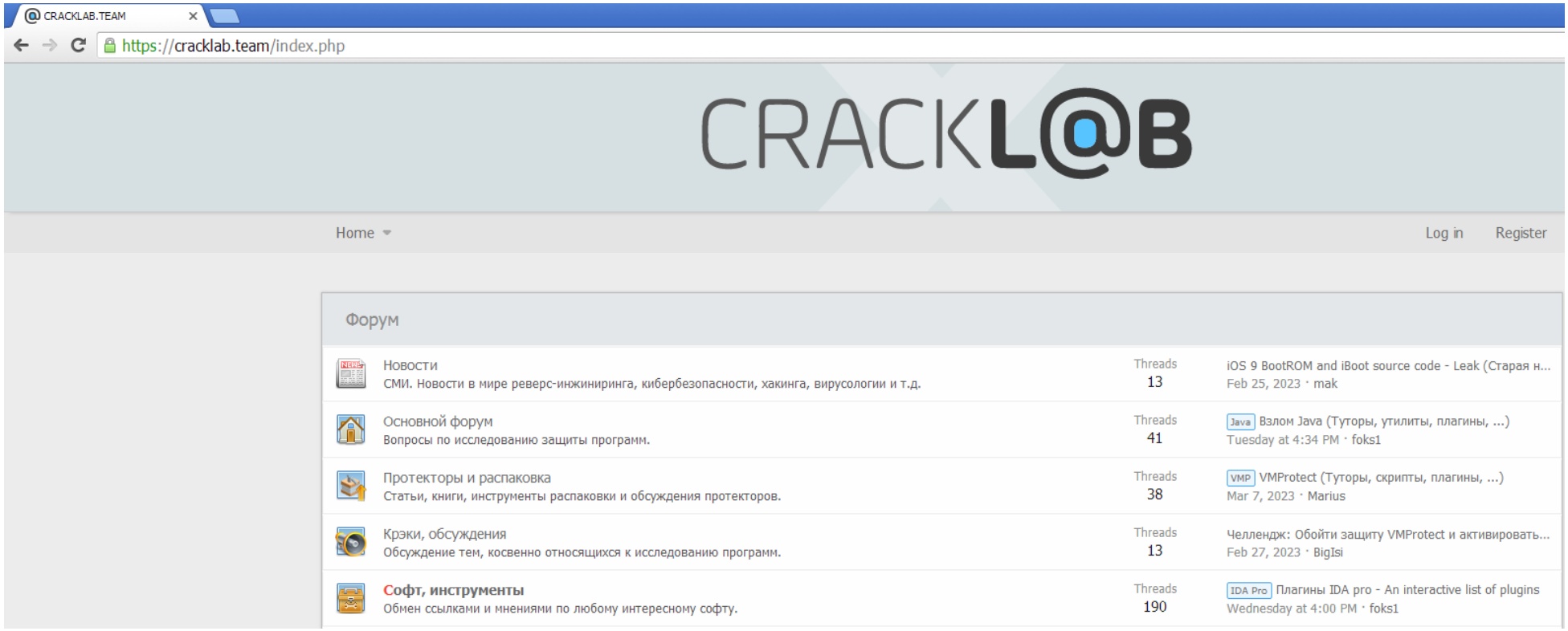
https://tuts4you.com/download.php?view.2090

https://cracklab.team/PAunlock/

https://github.com/Blaukovitch/80_PA/releases

https://www.reddit.com/r/Piracy/comments/42nt1h/what_is_this_crack_securom_denuvo/

securom80pa(at)gmail.com



- Hacking News
- Protectors and their unpacking
- Discuss your projects
- Popular hacking tools and utilities
- Hacking Q&A
- Technical Documentation
- Electronics & Cryptography

UNLOCK CODE	Game
KF3HC-7UZJ9-U3BDV-MP4QD-8B...	Epic Mickey 2: The Power of Two
KF3HC-7UXKY-CSL8G-N4SSK-RM...	Grand Theft Auto IV
CH746-RKFD5-KCZTP-8FLP4-2J...	Epic Mickey 2: The Power of Two
CH746-RKHM3-TTK5V-ZLN8A-27...	Grand Theft Auto IV

«Tiberiumny reversing»

(C) 2011-2026. ELF

