



---

## Ionworx ICE License. A Closer Look Inside

anorganix of ARTeam

March 2006

1.	Brief Introduction .....	2
2.	Things needed to get started .....	3
3.	Tomahawk Gold 3.0 .....	3
4.	getStarted!XP 4.5.6 .....	5
5.	LQKKnC 2.0.0.23 .....	9
6.	Conclusions .....	9
7.	Greetings .....	9

### Keywords

ICE License, Delphi, TomahawkGold, getStarted!XP, LQKKnC, OllyDbg, PExplorer, PEiD



### 1. Brief Introduction

ICE License is the strongest and most complete protection available today for Borland Delphi and C++ Builder. It is a new and innovative licensing protection system for Borland developers wishing to integrate copy protection and maximum security into their applications. With ICE License you can create "Trial Editions" of your software. Key generation is prevented using sophisticated asymmetric encryption technology. It is regularly evaluated by professional developers around the world. Their findings repeatedly demonstrate that this tool has many advantages over other copy-protection systems.

ICE License Overview :

- Antidebugging & Antimonitors Protection (protect against Code Tracing)
- Sophisticated Asymmetric Encryption Technology (prevent against Key Generator)
- Advanced Digital Authentication to prevent against patching
- Advanced Code Encryption Protection to prevent against cracking or dumping
- Transfer License to another PC (Trial License or Full License)
- Create a Key Generator for your partner or distributor
- Detects backdating or demo reinstallation to gain additional usage
- AES Rijndael Encryption functions to protect your data
- Cryptographic hashing MD5 to lock licenses to your software
- Reliable Machine Locking Protection (based on manufacturer's information)
- Network LAN Control Protection by TCP/IP Protocol (in Trial or Full Mode)
- Encrypt & Hide Application Strings to provide a high level of security
- Prevent and Control your LicenseKey against illegal exportation - New !
- Using compact LicenseKeys to register with ActiveKey for increased security
- Add Extra Information inside License (in Trial or Full Mode)
- View LicenseKey to see details about any License
- Create custom URL links to directly access your ordering web page
- Invisible software based protection requiring no dongles
- Create evaluation copies of your software by days, number of uses, hours or a set date
- Compatible with Windows 95, 98, Me, NT, 2000, XP (with SP2)
- Support Windows XP SP2 (from ICE License v2.09)

Find out more at:

<http://www.ionworx.com/IceLicense.html>



## 2. Things needed to get started

Required Tools	
<ul style="list-style-type: none"><li>» OllyDbg</li><li>» PEiD 0.94</li><li>» PE Explorer (or other resource editor)</li></ul>	

  

Target Applications & Protection	
<ul style="list-style-type: none"><li>» Tomahawk Gold</li><li>» getStarted!XP</li><li>» LQQKnC</li><li>» TlceLicense for Delphi</li></ul>	<ul style="list-style-type: none"><li><a href="http://nativewinds.montana.com/software/tomahawk.html">http://nativewinds.montana.com/software/tomahawk.html</a></li><li><a href="http://www.computentsystems.de/software.htm">http://www.computentsystems.de/software.htm</a></li><li><a href="http://chello.tucows.com/files3/lqqknc_2_trial.exe">http://chello.tucows.com/files3/lqqknc_2_trial.exe</a></li><li><a href="http://www.ionworx.com">http://www.ionworx.com</a></li></ul>

## 3. Tomahawk Gold 3.0

Our first target is called Tomahawk Gold, packed with UPX and using the ICE License protection system. Soon after opening the program we notice that we are dealing with a trial version with some disabled functions. See below:

About-Box	
User Name:	<b>Trial License</b>
User Company:	<b>Evaluation</b>
Product Name:	<b>Tomahawk</b>
Vendor Info:	<b>NativeWinds / Jack D. Lewis</b>
Creation Date:	<b>2005/11/23 @ 1:24:47 AM</b>
License Expiration:	<b>30 days</b>
Software Status:	<b>Trial</b>
Hardware Locked:	<b>No</b>
Network Protection:	<b>No</b>
Network User Allowed:	<b>No</b>

Now we open the target wit PEiD and see that it's compressed with "UPX 0.89.6 - 1.02 / 1.05 - 1.24 (Delphi) stub -> Markus & Laszlo". Notice that KANAL reports only "CRC32 [poly]". Easy enough, let's start unpacking. I will not go over unpacking UPX in detail. There are some very good tutorials about this topic. Shortly, fire up Olly and load our target.



## Ionworx ICE License. A Closer Look Inside

We are here:

Unpacking UPX with Olly		
00C75340	> 60	PUSHAD
00C75341	BE 00209A00	MOV ESI,Tomahawk.009A2000
00C75346	8DBE 00F0A5FF	LEA EDI,DWORD PTR DS:[ESI+FFA5F000]
00C7534C	C787 10B75E00 5F8>	MOV DWORD PTR DS:[EDI+5EB710],714A815F
00C75356	57	PUSH EDI

Press F8 to pass the PUSHAD (notice the ESP register is now highlighted). Select “Follow in Dump” and set a HW BP (Byte) on the first char. Hit F9 and the program breaks on a JMP. After that, push F7 to enter it and we land at the OEP. Let’s dump...

Reaching the Target’s OEP		
009B4F2C	55	PUSH EBP
009B4F2D	8BEC	MOV EBP,ESP
009B4F2F	83C4 F0	ADD ESP,-10
009B4F32	53	PUSH EBX
009B4F33	B8 FC3A9B00	MOV EAX,Tomahawk.009B3AFC

OK, so now we have unpacked the program. Load it in PEiD (again) and analyze it with KANAL. Wicked stuff we find...

PEiD’s KANAL Analysis	
1 x	BASE64 table
2 x	CRC32
1 x	FGInt ElGamalDecrypt
3 x	MD5

Seems damn hard, but in fact it’s easier than you imagined. Let’s open our target with PEXplorer and go to “Resources Viewer/Editor”. Expand the “RC Data” section and start browsing thru the forms.

At some point we reach “TFORM3” and by double-clicking on it, we notice the components it holds... look closer – the last one is “TIceLicense”. This is a very big achievement for us, because now we are dealing with some easy stuff. If we take a closer look at the TIceLicense component we will see the following:

PEXplorer’s Initial View	
OnAppKeyIncorrect	IceLicense!AppKeyIncorrect
OnTrialExpired	IceLicense!TrialExpired
OnLicenseInvalid	IceLicense!AppKeyIncorrect
OnLicenseFileError	IceLicense!AppKeyIncorrect
OnLicenseInfo	IceLicense!LicenseInfo
OnTrial	IceLicense!Trial
OnRegistered	IceLicense!Registered
OnTrialProgress	IceLicense!TrialProgress



Now it's obvious... We just need to do some replacements (with PExplorer). Your events should now look like this:

PExplorer's Modified View	
OnAppKeyIncorrect	IceLicense1Registered
OnTrialExpired	IceLicense1Registered
OnLicenseInvalid	IceLicense1Registered
OnLicenseFileError	IceLicense1Registered
OnLicenseInfo	IceLicense1Registered
OnTrial	IceLicense1Registered
OnRegistered	IceLicense1Registered
OnTrialProgress	IceLicense1Registered

After we save, let's test...

The program starts, and after a second or two it exits... hmm we did something that bothered him. Just open the file "Tomahawk.key" with Notepad (**do not delete it**), clear all text found inside and save it.

Why we do this?

Because we modified it earlier ("OnLicenseInvalid >> IceLicense1Registered" and "OnLicenseFileError >> IceLicense1Registered"). Now the program ignores the bad content of the Key file. **If you do not clear the Keyfile, the program will remain unregistered!**

Open the program and go to the about-box... yup, we did it!  
ICE License defeated!

## 4. getStarted!XP 4.5.6

A newer version of ICE License is a little bit different, and I say this because the authors removed most of the public events to improve protection. So we have to use a hybrid-method to defeat this version.

We start by analyzing our target. So, open PEiD and find out the target is compiled using "Borland Delphi 6.0 - 7.0". If you select the KANAL plugin, you will discover the following (again, do not worry because we will bypass all this crap to register the program):

PEiD's KANAL Analysis	
1 x	BASE64 table
2 x	CRC16 / CRC32
1 x	FGInt MontgomeryModExp
1 x	MD5
2 x	Rijndael

Now it's time to open our target in PExplorer. After a little browsing between the resources we arrive at one called "TFORM1".



## Ionworx ICE License. A Closer Look Inside

This resource is very interesting, because it contains the famous “TiceLicense” component. Double-click on it and take a closer look:

PE Explorer's Initial View	
CodeEncryption	True
AntiDebugging	True
AntiMonitoring	True
OnExeModified	IceLicense1ExeModified
OnLicenseTrial	IceLicense1LicenseTrial

My approach doesn't require debugging of the target, but if you want to explore / patch it using Olly, please make sure to disable the protections like below:

PE Explorer's Initial View	
CodeEncryption	False
AntiDebugging	False
AntiMonitoring	False
OnExeModified	nil
OnLicenseTrial	IceLicense1LicenseTrial

As you can see it doesn't contain “OnRegistered” event anymore, so we need another approach to our problem. Before continuing, download ICE License (trial) package from Ionworx's site, because we will need it. I will explain why...

After installing ICE License, run the file called “ICE License Manager.exe”. As you will notice there is no trial limitation, just a nag-message that is not important. Now open it in PEiD and select “PEiD Generic Unpacker” plugin. Press “Unpack” and you should have a file called “ICE License Manager.unpacked.exe” in the same location as the original one. The file isn't working (thanks to Shub for the partial dumps tutorial) but we don't care because we are just interested in exploring it's resources.

So, open it with PE Explorer and go to “TFORM1”. Notice that it (obviously) contains a “TiceLicense” component that isn't in trial mode. So we just need to get the values from it and put them in the trial “TiceLicense” component found inside “getStarted.exe”. We don't know what the encrypted string mean, but we know that they will turn our program into registered mode...

ICE License Manager.unpacked.exe	
ICE_Data1	tA3bpy6+aZPPwqsempwSxg==
ICE_Data2	Ffpb5POWZUIJ+isSre3nZ3eif12IRJBU4H0O9MUOVmg=
ICE_Data3	AaSjTag+JmCuYNLHRRrPOVw==
ICE_Data4	AaSjTag+JmCuYNLHRRrPOVw==
ICE_Data5	ZoWn69d1asyJ4DLSisDfhqHjMNOxn12IzdxlmX4b9yA=
ICE_Data6	LM1/VCKlnkSdi8Pd2Ejj/mYqf0qERlCpmVitejMkj68=
ICE_Data7	ilnmRK5a2otHIER+0oniUGuWEMsGg1zyR8s4hsHQLxk=
ICE_Data8	AaSjTag+JmCuYNLHRRrPOVw==
ICE_Data9	AaSjTag+JmCuYNLHRRrPOVw==

Now we need to replace the values found in “getStarted.exe” with the ones in the table above to remove the trial. Apparently the “ICE\_LProtection” field doesn't affect the program, but if you are willing to experiment, try playing with it a little... Save your edited file and test it. Trial is gone!



### 5. LQQKnC 2.0.0.23

To defeat ICE License we will use a different approach. If you start the program, you will notice a 30-day trial nag... after this, the program is 100% functional. Let's analyze it using PEiD. Mine reports "UPX 0.89.6 - 1.02 / 1.05 - 1.24 -> Markus & Laszlo". Unpack it however you wish... I will not go into details with unpacking. I used UPX's "-d" switch to obtain the "virgin" exe.

It's time to open the file in Olly. Make sure to ignore all exceptions and hit F9 to run the target (if your Olly will close it's because of ICE protection; don't worry, it's lame... you can use my program xFile to hide the caption of your tools).

When the nag-screen appears, press F12 to pause the execution, and then Alt+K to bring up the Call Stack. Double-click on the last item and arrive here:

Olly View			
0048E390	/ \$	53	PUSH EBX
0048E391	.	83C4 E4	ADD ESP, -1C
0048E394	.	8BD8	MOV EBX, EAX
0048E396	.	8BD4	MOV EDX, ESP
0048E398	.	8BC3	MOV EAX, EBX
0048E39A	.	E8 41FFFFFF	CALL Cracked.0048E2E0
0048E39F	.	84C0	TEST AL, AL
0048E3A1	.	75 09	JNZ SHORT Cracked.0048E3AC
0048E3A3	.	8BD4	MOV EDX, ESP
0048E3A5	.	8BC3	MOV EAX, EBX
0048E3A7	.	E8 E4080000	CALL Cracked.0048EC90
0048E3AC	>	83C4 1C	ADD ESP, 1C
0048E3AF	.	5B	POP EBX
0048E3B0	\.	C3	RET

Now, put a breakpoint (F2) on the RET at 48E3B0 and hit F9. Olly will break on the RET. Remove the breakpoint (F2 again) and press F8. Scroll down a little bit until you arrive at this view:

Olly View			
0048B251	.	E8 B2220000	CALL Cracked.0048D508
0048B256	.	C3	RET
0048B257	^	E9 6095F7FF	JMP Cracked.004047BC
0048B25C	^	EB EE	JMP SHORT Cracked.0048B24C
0048B25E	.	33C0	XOR EAX, EAX
0048B260	.	5A	POP EDX
0048B261	.	59	POP ECX
0048B262	.	59	POP ECX
0048B263	.	64:8910	MOV DWORD PTR FS:[EAX], EDX
0048B266	.	68 7BB24800	PUSH Cracked.0048B27B
0048B26B	>	8D45 E0	LEA EAX, DWORD PTR SS:[EBP-20]
0048B26E	.	E8 E59BF7FF	CALL Cracked.00404E58
0048B273	.	C3	RET
0048B274	^	E9 4395F7FF	JMP Cracked.004047BC
0048B279	^	EB F0	JMP SHORT Cracked.0048B26B
0048B27B	.	8B45 F8	MOV EAX, DWORD PTR SS:[EBP-8]
0048B27E	.	5E	POP ESI
0048B27F	.	5B	POP EBX
0048B280	.	8BE5	MOV ESP, EBP
0048B282	.	5D	POP EBP
0048B283	.	C3	RET



## Ionworx ICE License. A Closer Look Inside

Again, put a breakpoint on the RET at 0048B283 and hit F9 to run th app. Click on “Quit” or “Continue” button to make Olly break. Remove the breakpoint and hit F8.

Olly View		
00757821	. 55	PUSH EBP
00757822	. 68 87787500	PUSH Cracked.00757887
00757827	. 64:FF30	PUSH DWORD PTR FS:[EAX]
0075782A	. 64:8920	MOV DWORD PTR FS:[EAX],ESP
0075782D	. A1 5CFE7600	MOV EAX,DWORD PTR DS:[76FE5C]
00757832	. 8B00	MOV EAX,DWORD PTR DS:[EAX]
00757834	. 8B10	MOV EDX,DWORD PTR DS:[EAX]
00757836	. FF92 EC000000	CALL NEAR DWORD PTR DS:[EDX+EC]
0075783C	. 3D 00040000	CMP EAX,400
00757841	. 74 21	JE SHORT Cracked.00757864

This is it, the nag-screen is called at address 00757836. If we look closer we see a CMP EAX, 400. If we NOP the CALL we also have to force a jump at 00757841 (because EAX will not be equal to 400). So instead of patching in 2 places, just change the CALL to MOV EAX,400. This way the nag is killed and the jump is taken.

Save the new file, load it into Olly and press F9. A nasty message says that “Certificate is missing. Please contact support@lqqknc.com”. Press Ctrl+F2 to restart the program in Olly. Do a right-click and select “Search for >> All Intermodular Calls” and search for the string “Certificate”. When it’s found, double-click on it and scroll up to the beginning of the function, where you will place a breakpoint. There will be 2 occurrences of the string in the program so remember to see both. You should have 2 breakpoints set on the following addresses:

Olly View		
0074AE88	\$ 55	PUSH EBP
0074BAC4	\$ 55	PUSH EBP

Now press F9 and Olly will break here:

Olly View		
0074BAC4	\$ 55	PUSH EBP
0074BAC5	. 8BEC	MOV EBP,ESP
0074BAC7	. 6A 00	PUSH 0
0074BAC9	. 6A 00	PUSH 0
0074BACB	. 53	PUSH EBX
0074BACC	. 56	PUSH ESI

The problem is the CALL at the line 0074BB22 (CALL 006D0348). Instead of using NOP on this CALL we will make it return. Press Ctrl+G and type the address “006D0348”. Now replace the “PUSH EBP” with a “RET”. Save you file and press F9 to run (ignore the BPs).

Now we just got rid of one message, and another will pop up saying that “Your license is missing. Please contact support@mark.net”. OK, restart and search for the string “license is missing”.





All you have to do is to change the JNZ at 0074AEF8 to JMP. And now for the trial check. Just set your date in the future, and when you will run the program it will give you a nice nag saying “Your trial is over! Please register!”. Press F12 to pause the execution and follow the steps described for the first nag (using the Call Stack). You will arrive here:

Olly View		
0049416F	> \8B45 F4	MOV EAX, [LOCAL.3]
00494172	. 8B10	MOV EDX, DWORD PTR DS:[EAX]
00494174	. FF92 EC000000	CALL NEAR DWORD PTR DS:[EDX+EC]
0049417A	. 8945 F8	MOV [LOCAL.2], EAX
0049417D	. 33C0	XOR EAX, EAX
0049417F	. 5A	POP EDX

To remove the nag, just NOP the CALL at 00494174. Just to be sure, also search for “Reminder”. One is found at 0074BCA6 and one at 0074C0A6. In both cases, change from JGE to JMP.

Save your file and test it... it works like a charm!

Note: ICE License also incorporates some “machine-restrictions” and “network-control” options... so if you want to move the program to another computer or use it in a network, it might be necessary to check for these limitations too.

## 6. Conclusions

Well, this is the end of this story, I hope all the things said here will be useful to understand future versions of ICE License. I suggest as usual to use this material for learning purposes only, and not for cracking programs. **Thanks for reading this tutorial!**

### Disclaimer

All the code provided with this tutorial is free for public use, just make a greets to the authors and the ARTeam if you find it useful. Don't use these concepts for making illegal operation, all the info here reported are only meant for studying and to help having a better knowledge of application code security techniques.

## 7. Greetings

*Thank you Pilli for your support! You are the best!*

[ ARTeam ] [ EXETools ] [ all the RO scene ] [ bLaCk-eye ] [ vybez\_mR ]