Reverse Engineer*ZINE*

An Immortal Descendants Production
http://www.ImmortalDescendants.com

Issue 002 **October 1999**

Forward

Greetings. I got some very nice feedback on the first issue of this zine. Most people who commented seemed to really enjoy it, so thanks for your feedback, and I'm glad you enjoyed it! As usual, please feel free to e-mail me your comments and/or criticisms: Volatility@ImmortalDescendants.com

Enjoy!
Volatility

Scene News

IDA 4 Demo is out, featuring a tough 1024bit RSA protection. It's only a matter of time though.
Fravia's new fortress was hacked. He's keeping a copy of it if you care to see it.
Several new up-and-coming scene sites are being developed. Look for them to be featured in future issues.

Protection Of The Month

This month's protection is for Winimage v5.00.500 available at: http://www.winimage.com. If you frequent the warez scene :(, or if you do alot of CDR, you should already know what this program is. It extracts files from ISO images... nonetheless, we're just interested in the protection scheme :)

Finding your correct serial number in Soft Ice is quite easy, but you won't learn anything. Your goal should be to write a key generator. The only hint I'll give you, is BPX GetDlgItemTextA -- but then again, you could have figured that out :) As usual, an solutions/essays will be featured here, if you send them.

alpine's Solution - Making the program create it's own key generator (10/04/99)

Tool Of The Month

Since IDA 4 was just released, it's fitting to list this as this month's tool. If you haven't used IDA, you really should give it a try. It's very different from W32dasm, and provides some very unique features. Grab the demo at http://www.datarescue.com.

Commercial Stupidity

These stupid protections were submitted by Tornado... I'd suggest you don't waste your time downloading and cracking these, as you won't learn a thing. They're listed here for humor purposes, and so shareware authors can learn what NOT to do :) Serial numbers and such have been removed, and replaced with asterisks (*).

=-=

The Honor System (Not Too Alive And Well)

=-=

I just found a protection in a shareware program that is worthy to be mentioned as the most STUPID one I ever found (or do you just want to list the best?) ... ok not the complete protection is so stupid ... With this short info you can't get the BONUS UTILITY that is provided to registered users (download instructions by e-mail).

"Once you have registered, you may click the box below. This will disable the opening "Please Register" screen.
[ ] On my honor, I have paid for this software."

So I just clicked on the box awaiting a dialog box to pop up for entering my registration details ... but NO details needed? I just got:

"Thank you for registering Screen Paver. Your payment helps to maintain and continue development of this software. Click OK to continue as registered user, or click Cancel to continue unregistered (register later). OK CANCEL"

And guess what happend? ... yeah your're right ... after pressing OK the screen disappeard next time :) Well if you are willing to check out this stupidty ... go to http://tni.net/~mlindell/ScreenPaver.html "Written in C++ by Michael Lindell using Power++ Developer." I don't know Power++ Developer thing (I guess it's kinda template or functions for your programs) anyway stupid coder ... at least there should have been added a registration key or something of that kind. Hopefully next time I'll have a GOOD protection :)

TORN@DO
=-=

Hardcoded Stupidity

=-=

Dear Volatility,
I'm writing you this e-mail (again) for a MOST STUPID protection I've come accross ... the current one is very very stupid ... maybe u decide otherwise:

Net Executive Screen Savers: WWF 1.0.5.2517 http://www.netexecutive.net

This screen saver allows entry of a password to remove any expiration period or unlock more photos.

"If you have the unlocking password, please enter it now ________ This demo version expires in 12 days. OK CANCEL"

As I have assumes there was a universal unlock code ... I found it with SoftICE in less than 30 seconds :) Now for you I wanted to check out how silly the coders where and simply searched for password (unlockpassword) and found: UnlockPassword=****/******/****** That was very disappointing ... I made more fun :) Searching for ExpireDays ... and found ExpireDays=13 Even more stupidity in this program ... I don't want to go on (although it's fun) ... this is surely COMMERICAL STUPIDITY :)
TORN@DO
=-=

Link Of The Month

Last month's issue featured an interview with knotty of DREAD. Dread has recently aquired a new domain! Please update your bookmarks to http://www.sresrever.com. Reverse that you reversers! :)

Article Of The Month

No interview this month. I figure it's stupid to have an interview EVERY issue, as it could get boring, and I'd soon run out of people to interview! If you want to see an interview each issue though, let me know, and I'll consider it. This is a very interesting article about so-called "Quantum Cryptography". If feasible, it's applications could be seen in the reversing community in the future. This article shamelessly swiped from http://www.newscientist.co.uk.

Quantum Confidential

Photography: John Millar

Want to beat the hackers once and for all? As Simon Singh finds out, the enigmatic quantum world is about to make your secrets safe as houses.

IT COULD HAPPEN in a few months or a few years. But sooner or later, a mathematician could make a discovery that jeopardises international security, threatens the future of Internet commerce, and imperils the privacy of e-mails. Today's codes and ciphers are good, to be sure. But it is probably only a matter of time before they become useless.

With the coming of the information age, we rely ever more heavily on cryptography to protect us from snoopers, cyber-crooks and Big Brother. Some of today's most secure codes exploit the fact that while it is easy to multiply two prime numbers together, it is almost impossible to start with the answer and work out which two primes were used to create it. But the day a mathematical genius discovers a short cut for finding the hidden prime numbers, these codes will crumble.

What everyone is looking for is a new form of code, one that is truly unbreakable. That's where the quantum world comes in. Exploiting the strange uncertainties of quantum physics can give you a code that nobody--no matter how clever--will ever be able to crack.

That's the theory. The trouble comes when you try putting it into practice. When quantum particles interact with the large-scale world they tend to lose the delicate information they contain. This makes it fiendishly difficult to use them to send information over any sensible distance. Difficult, but not impossible. In the past few years, researchers have succeeded in sending quantum-encrypted messages tens of kilometres down optical fibres. Now the challenge is to find a way to send quantum-encrypted information through the air. This will open the way to fully secure global communications, beamed up to an orbiting satellite and forwarded to any place on Earth. It's a phenomenal technical problem, but this year researchers at the Los Alamos National Laboratory in New Mexico achieved a breakthrough that looks set to transform the way we keep our secrets.

Cryptographers often describe code scenarios in terms of a trio of characters called Alice, Bob and Eve. While Alice is trying to send a sensitive message to Bob, Eve is trying to eavesdrop. To keep her message secret, Alice has to encrypt it, and for this she can use a cipher known as a "one-time pad". Cryptographers have known about the one-time pad technique for decades and it is logically uncrackable. The encryption requires three separate stages. First, Alice transforms her message into a series of 1s and 0s. Second, Alice creates a key--a random series of 1s and 0s that is as long as the message. Third, Alice adds each element of the key to the corresponding element of the message, to create an encrypted text also made up of 1s and 0s; the only unusual adding rule is that 1 + 1 = 0. Finally, Alice sends the encrypted text to Bob.

This type of code is impossible to crack because each element of Alice's key is random. Even if Eve were to use computational brute force to try every possible key, she'd find that many of them made some sort of sense, and wouldn't know how to choose between the alternatives. Bob, on the other hand, has a copy of the key, and can decipher the message by simply subtracting the key from the encrypted text.

The one-time pad cipher is so called because each key used to be written on a separate sheet of a pad of paper. After being used once, the sheet was torn off and destroyed, leaving the new key on the next sheet ready to encrypt the next message. Despite being theoretically perfect, the one-time pad cipher suffers from several practical flaws, which have prevented its widespread use. Making random keys is a difficult task, and making a new one for each message is time-consuming. The real killer, though, is distributing the keys. After Alice has manufactured a random key, encrypted her message, and sent the encrypted text, she somehow has to get the key to Bob so that he can decrypt the message. She cannot send the key unencrypted because Eve will steal it, and she cannot encrypt it because she then has to tell Bob the key she used to encrypt the key that she used to encrypt the message.

The key-distribution problem was traditionally solved by employing trusted couriers to deliver the keys by hand, but this solution doesn't have much appeal in the age of satellite communications and e-mail. It is here that quantum physics comes to the rescue. In the early 1980s, Charles Bennett, an IBM researcher, and Gilles Brassard, a computer scientist at the University of Montreal, proposed that Alice and Bob should use individual photons to exchange their key. By operating at the quantum level, they argued, Alice and Bob could exploit the laws of quantum physics to protect the key.

Bennett and Brassard proposed using photons polarised in different directions to represent 1 or 0. If Eve tried to intercept the key, she would have to measure the photons, which would effectively mean absorbing them. To avoid being spotted, Eve would have to retransmit the photon to Bob. However, because of the strange way that quantum particles work, Eve does not always measure the same polarisation that Alice sent. That in turn means that she cannot be sure that she is retransmitting the correct orientation. Thus Eve's interception will inevitably affect the transmission of the key, and Alice and Bob should be able to spot this, discard the key, and try again with a new one.

The system is perfect, apart from one problem. If Eve cannot accurately read the key, then how can Bob? In 1984, Bennett and Brassard were chatting on the platform at Croton-Harmon station in New York state, near IBM's Watson Laboratories in Yorktown Heights, when they hit on the answer. Waiting for the train that would take Brassard back to Montreal, they invented the first workable form of quantum cryptography. The Bennett- Brassard communications protocol requires the use of four polarising filters for Alice and four for Bob, but it was superseded in 1992 by a simpler system that requires only two filters each.

It works like this. Alice needs to send a key to Bob, which he can then use to decipher a future coded message. To do this, Alice starts with two polarising filters oriented at 0 degrees and +45 degrees, representing 0 and 1 respectively. Bob has two similar polarising filters oriented at 90 degrees and -45 degrees. For the key, Alice sends Bob a string of randomly polarised photons representing 1s and 0s. Bob then tries to measure the polarisation of each photon by randomly switching between his two filters.

A photon striking a filter oriented in the same direction will always pass through. Conversely, a photon striking a filter oriented perpendicularly will never pass through. But a photon hitting a filter that is diagonal to its own orientation is in a quantum quandary, with a 50:50 chance of passing through or being blocked.

Suppose Bob chooses his -45° filter to measure a photon from Alice, and no photon passes through. He cannot know whether Alice sent a +45° photon (meaning 1), which is always blocked, or if she sent a 0° photon (meaning 0), which is only sometimes blocked. If a photon does pass through his filter, then he is in luck: he can be sure that Alice sent a 0° photon.

This means that Bob knows that if a photon passes through his -45°filter, Alice must be sending him a 0. Similarly, if he uses his 90° filter and the photon passes through, then Alice must have sent a +45° photon (see Diagram, p 32).

So when Alice sends polarised photons to Bob, he will be able to establish with certainty the bit value of a fraction of them. Alice could send a series of a hundred photons, each one polarised at random, while Bob randomly switched between his filters. Typically, three-quarters of them would be blocked, but Bob would know the bit value for the lucky minority that got through. Bob could then call Alice on the telephone and tell her exactly which 25 photons he received. These would form the key for encrypting a subsequent message (see Diagram, p 33).

Filtered out

Although Bob tells Alice which photons he correctly measured, he does not say which filter he used to measure them. So even if Eve overhears the telephone conversation, she gains no information about the composition of the key.

And, crucially, if Eve tries to intervene at an earlier stage by intercepting the photons on their way to Bob, then her presence becomes apparent. Suppose that Alice sends a 0° photon, representing a 0 bit, and Eve measures it using a -45° filter. If the photon is blocked, Eve does not know if this is because the photon was +45°, and so stood no chance of passing through, or because it was at 0° and she was unlucky. Eve might take a guess that it was a +45° photon, and create and transmit such a photon onto Bob. If Bob measured it using his 90° filter, the photon might pass through--and if it did he would incorrectly interpret Alice's photon as representing a 1 bit.

Bob's misinterpretation can be used to expose Eve's nefarious interception. To see if Eve has been listening, Alice and Bob check for errors. After establishing a tentative key, they pick some of the bits at random and declare their values over the telephone to see whether they agree. If there is any discrepancy, they assume that Eve has been eavesdropping and they abandon the key and start again. If there is no discrepancy, they assume that it is safe to use the key as the basis for encrypting a message--having first discarded those bits that they disclosed during the error-checking procedure.

There is always the possibility that Eve intercepts a photon and guesses correctly when she retransmits it. If such a photon is used as part of the error-checking procedure, then no error appears, and Eve's presence is not betrayed. But as Bob and Alice check more and more bits, Eve's chances of avoiding detection become vanishingly small.

Once the key has been sent successfully, Alice uses it to encrypt her message. She can then send the message by phone, pigeon post or whatever. Safely encoded, it can't be deciphered by anybody but Bob, even if it's there for all to see.

It was not until 1989, five years after Bennett and Brassard invented quantum cryptography, that they tested it experimentally. One computer, Alice, sent a stream of photons through 32 centimetres of air to a second computer, Bob. Bennett and Brassard had successfully transmitted the world's most secure key.

Other experimenters soon began to design systems that operated over more useful distances. The crucial technical problem is maintaining the polarisation of the photons. If this changes during transit, Alice and Bob's error-checking procedure will find discrepancies even if Eve is not eavesdropping, so no valid key will emerge.

One way round this is to send photons down optical fibres, which conserve the polarisation. Already, this approach has allowed quantum-encrypted messages to be sent over significant distances. In 1995, researchers at the University of Geneva successfully sent a message down an optical fibre to the town of Nyon, more than 20 kilometres to the north. This year, researchers at Los Alamos established a new record when they sent a quantum key through a 48-kilometre optical fibre--long enough to set up a network between neighbouring branches of a bank say, or government offices. But extending the technology any further is more problematic, because individual photons struggle to survive the journey through the fibres without being absorbed. Over distances of hundreds or thousands of kilometres, the signal would dwindle to nothing.

The ideal solution would be to find a way to send quantum keys up through the air to waiting satellites. The Quantum Information Team at Los Alamos, led by Richard Hughes, is the world leader in such "free-space quantum cryptography". For the past two years, the group has been steadily overcoming the technical difficulties and extending the transmission distances step by step.

Ultimately, they want to be able to fire individual photons to hit a satellite's receiver, which is only a few centimetres across and orbits at an altitude of 300 kilometres. The photons must pass through the atmosphere without being absorbed--so that the signal is not simply lost--and they must not change their polarisation.

It's easy enough to make sure that the photons are not absorbed. You just have to choose a wavelength that the molecules in the atmosphere ignore. Hughes's team has opted for 770 nanometres. Longer wavelengths also pass through the air unscathed, but are more susceptible to turbulence, which changes the local refractive index of the air and thus twists the orientation of the photon's polarisation. Turbulence typically occurs on a scale of tens of centimetres, so 770 nanometres is short enough to avoid this.

That still leaves plenty of other problems. For instance, as the satellite tries to detect Alice's photons, there is the risk of being swamped by background photons, either coming directly from the Sun or reflected from the Earth or Moon. To prevent this, the Los Alamos group has designed a highly directional receiver that only picks up photons arriving from Alice's direction. It also includes a filter to ensure that only photons of the correct frequency are accepted.

To exclude any remaining extraneous photons that happen to come from the right direction with the right frequency, the detector only accepts photons that arrive during a time window of 5 nanoseconds each microsecond. The window has to be open when Alice's photon arrives.

But this causes another problem--and again it's turbulence that is at the root of it. Even if it doesn't change the polarisation of the photons, turbulence does affect how fast they travel. This leads to "jitter"--a continual variation in the journey time. To compensate for jitter, a pulse of light is sent 100 nanoseconds ahead of each photon. This timing pulse is affected by the atmosphere in exactly the same way as the photon that follows. So whenever a pulse arrives, the satellite knows that the photon will be coming 100 nanoseconds later and times the opening of the window accordingly.

Turbulence causes another headache too. Changes in refractive index cause the beam to wander so that it misses the satellite's antenna. To keep the

photon beam on course, Alice monitors the feeble reflections from the timing pulses, and uses the information to steer the photon beam.

Earlier this year, Hughes set a new record for quantum cryptography through air when he exchanged a key across 500 metres. Bob, the receiver, was equipped with a 3.5-inch-diameter telescope. Each incoming photon encountered a beam splitter, which randomly reflected or transmitted the photon, steering it towards one type of filter or the other.

Afterwards, Alice and Bob used an insecure Ethernet link to check for errors in their key. Because there was no Eve attempting to intercept the key, there should have been no errors. In fact, background photons, detector noise and misalignment introduced an error rate of 1.6 per cent, but this isn't too serious. If Eve had been listening in, she would have caused an error rate more like 25 per cent, so Alice and Bob can still be confident that their key is secure.

So how close is practical free-space quantum cryptography? At first sight there's a big difference between the 500 metres Hughes sent his quantum cryptographic key and the 300 kilometres needed to reach a communications satellite. But Hughes is closer to his goal than these figures suggest. His photons travelled horizontally, at ground level, where the air is dense and fluctuations are greatest. Hughes estimates that transmitting a quantum key 2 kilometres horizontally would be equivalent to reaching a satellite in low-Earth orbit. He plans to try a 2-kilometre demonstration later this year. Then, within two years, he hopes to conduct a quantum cryptographic exchange with an actual satellite.

If Hughes's experiments go to plan, global satellite communications could be protected by secure quantum cryptography within a decade. In the meantime, optical fibres will allow communications on a much smaller scale. Even now, it would be possible to build a quantum cryptographic optical fibre link between the White House and the Pentagon. Perhaps there already is one.

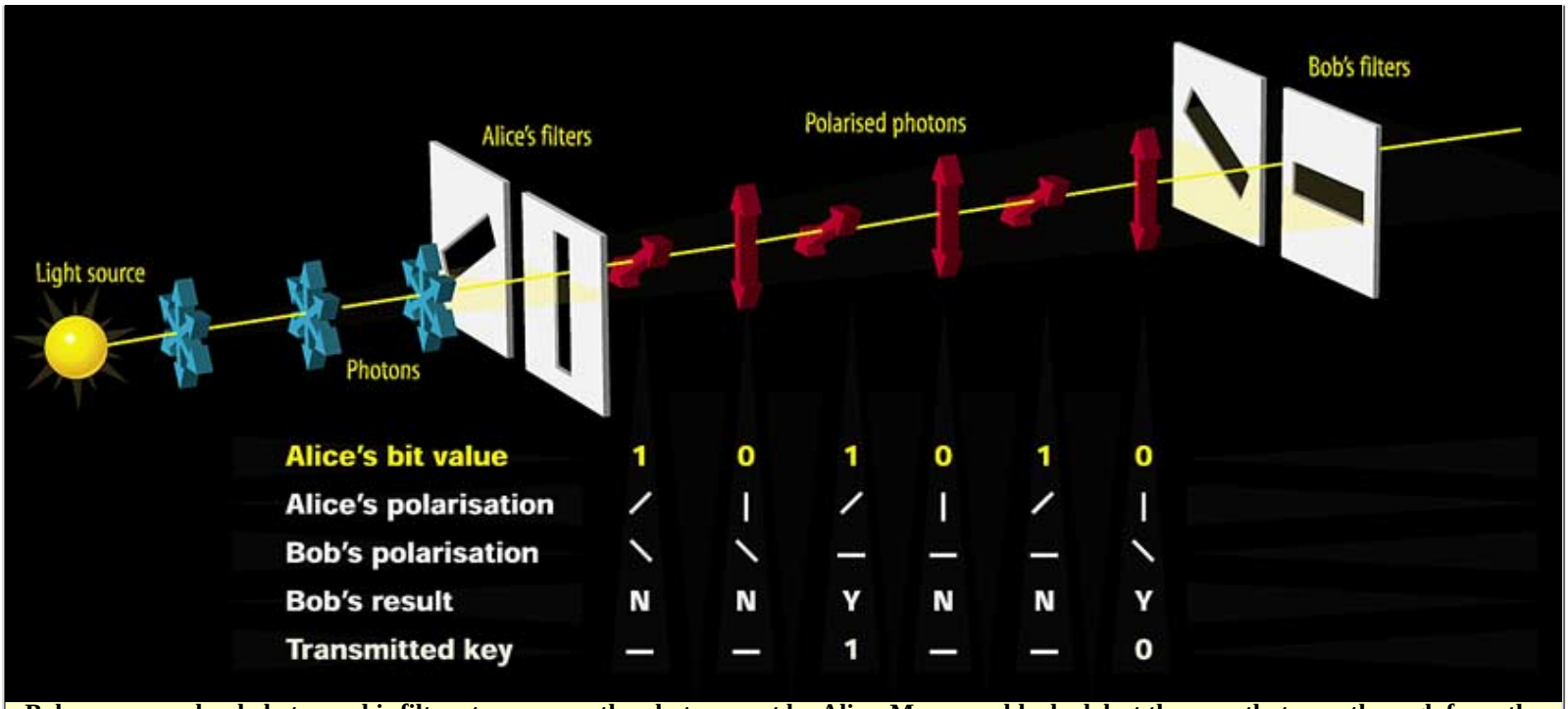
Simon Singh is a freelance writer based in London, whose **The Code Book--the science of secrecy from Ancient Egypt** to quantum cryptography was published this month by Fourth Estate. He is also the author of **Fermat's Last Theorem**

Graphics:

Quantum cryptography: Photon transmission (A)

Photons that pass through Bob's 90 degree and -45 degree filters must have started off as 1 and 0 respectively

Quantum cryptography: Photon transmission (B)



Bob swaps randomly between his filters to measure the photons sent by Alice. Many are blocked, but the ones that pass through form the key.

Credits, Greetings

Thanks again, for checking out this issue. I hope you've found it helpful, and interesting. Please don't hesitate to send me your comments. Any additions for the next issue will be MUCH appreciated.

Credits and thanks for this issue go to: alpine, Authors of Winimage, DREAD, Tornado, New Scientist.

Personal greetings fly out to: alpine, knotty, Latigo, LaZaRuS, Lord Soth, Lucifer48, Neural, Tornado, WarezPup, Yoshi, and everyone I forgot (probably MANY)

Copyright 1999 Volatility and the Immortal Descendants

Beating the program with it's own weapons

hello all!

Welcome to this short solution on how to get the serialnumber with the programs own routines.The method i'll describe here isn't used very often, although it has the same effect as a keygen but is easier to do.

part 1
theory

We're gonna wait till the prog calculates our serial, then move the calculated serial to another place in memory, otherwise it would get overwritten by some other stuff and finally we patch the msg-box to tell us the serial instead of the "wrong serial....." message. that's all. Easy eh?

part 2
war

Due to the fact i wanna shorten up my article i'm not gonna show you how i found the places. Just the facts:

a.) we need the place where the serial is converted,so it can be shown in a msg-box, and moved to a memory location (! that's not the place where it's calculated!).
The prog uses wsprintfa to do this.

That is the call to the routine:

```
0043480D 8D8500FEFFFF            lea eax, dword ptr [ebp+FFFFFE00]
00434813 57                      push edi
00434814 50                      push eax
00434815 E862FFFFFF              call 0043477C
```

the first line is the pointer to the textbuffer where the converted serial number gets written to ([ebp+FFFFFE00] = [ebp-200])
As i said before we've to change that pointer else the serial will get overwritten by some crap.so take softice break on that (0043480D) and assemble it to 'lea eax, dword ptr [ebp-150]'
Remember that location [ebp-150] , we'll need it again.

b.) we need the place where every thing gets pushed for the msg-box:

```
0041FC44 8D8580FBFFFF            lea eax, dword ptr [ebp+FFFFFB80]
0041FC4A 56                      push esi
0041FC4B 50                      push eax
0041FC4C FF7508                  push [ebp+08]
0041FC4F E873FFFFFF              call 0041FBC7
```

first line: lea.... puts a pointer to the string" Registering information is invalid" or something like that. Guess what we do
Yes we change that pointer to the location we've put the real serial to.
Important thing: ebp has changed its value (4bytes)

so mov to that line (lea..) and assemble it to lea eax, [ebp-154]
press f5 and you'll get the a nice msg box telling you the real serial.
Now if you wish make it static with an hexeditor and enjoy life.
Don't forget to buy the proggie if you wanna use it,
coz shareware authors are our source.

NOTE: There are two levels of protection. You can choose to register as "Standard" or as "Professional". Using the same method as above, you just need to change the other location.

greetings to all i know
Special thanks,in no order to Volatility,Lord Soth,Lucifer48,Acid_burn,
WarezPup,icecream,Tornado,RevX,Lazarus,....

contact me through email: alpine@ImmortalDescendants.com
or visit us at: www.ImmortalDescendants.com