```
-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_

    (NNNNNN. (NN)                  (NN)     (NN)  (NN)         ___                NNN    NNNN)   JNN   .NN)
    (NNNNNNN (NN)   ___.     ___   (NN) ___ (NN)  (NN)  .___   NNN    .___    .__ NNN   JNNNN)(NNNNN NNNNN)
    (NN) NNN (NN) (NNNNNN  (NNNNN) (NN)(NN) (NN)  (NN) JNNNNN.(NNNN) NNNNNN. (NNNNNNN  .NN4NN)`"4NNN ""NNN)
    (NNL_NNN (NN) NNN (NN) NNN (NN (NN)NNN  (NN)  (NN) NNN (N)`NNN"`(NN) NNN (NN) NNN  JNF(NN)   NNN   (NN)
    (NNNNNN) (NN) NNN.JNN) NNN (NN (NNNNN`  (NN)  (NN) NNNL/"` NNN  (NNNNNNN (NN) NNN (NN`(NN)   NNN   (NN)
    (NN) NNN (NN) .NNNNNN) NNN     (NNNNN)  (NN)  (NN) `NNNNN  NNN  (NNNNNNN (NN) NNN NNN_JNNL.  NNN   (NN)
    (NN) NNN (NN) NNN`(NN) NNN (NN (NN)NNL  (NN)  (NN) __`NNN) NNN  (NN) NNN (NN) NNN NNNNNNNN)  NNN   (NN)
    (NNLJNNN (NN) NNN (NN) NNN (NN (NN)(NN. (NNNNN(NN) NN (NN) NNN  (NN) NNN (NN) NNN NNNNNNNN)  NNN   (NN)
    (NNNNNNF (NN) NNNNNNN) (NNNNN` (NN)`NNL (NNNNN(NN) 4NNNNN` NNNN) NNNNNN) (NNNNNNN     (NN)   NNN   (NN)
    `""""""  `""`  ""``"""`  """   `""` """ `"""""`""`   """    """`  `"""    `"" """     `""`   """   `""`
-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_-_
```

"BLACKLISTED 411 .NET"

Edition 4

01.16.2006

-=-

Table of Contents

-=-

[1] ==Introduction: 2006 Updates==

Welcome to 2006

With the onset of 2006, a lot of things are changing at Blacklisted411. First off, we finally gave the printed edition of Blacklisted411 a NEW and improved look. The front cover will no longer be a generic template, but rather a FRESH design for every edition. We also started production of a membership card which gives subscribers benefits at conferences and retailers. In addition to the membership card and design changes, new bumper stickers, shirts, hats and dog tags are in the works. We also are working on creating a “Hacker License/ID” sticker which we feel will be a great source of entertainment for the hacking community.

Edition Notes

This .NET edition may seem just a tad bit smaller since a lot of our writers are busy and the overlapping release of the printed edition generally causes things to become hectic. Most notably in this edition is our research on icemodchip.com (A controversial website that has split the Xbox modding community recently). With this article, we hope to give our readers a better understanding of what icemodchip.com really is. Of course, this research will also be released in a separate PDF for distribution purposes. Along with the icemodchip.com article, Israel Torres has provided an excellent article titled “Hacking Cryptograms 101”. As always, I am still looking for good writers willing to help out in any way.

ENJOY

[2] ==Edition Quote==

“All warfare is based on deception”
Sun Tzu

[3] ==Articles==

[a]

Hacking Cryptograms 101

Quid Pro Quo

By Israel Torres <israel@israeltorres.org>

Welcome to the third lesson in hacking cryptograms. If you haven't read the other two I implore that you do at some point in time. The main purpose of this column is to explain an approach to solving the previous cryptogram (*available at the end of each Blacklisted411.NET edition*). I also provide source code so that it can be done programmatically at your leisure so you may learn from playing with it yourself. I love crypto in all its glory and enjoy playing with it as much as I can, whenever I can. For the last issue I was asked to make two cryptograms; one for a beginner, and one for those with intermediate to advanced crypto solving skills. After thinking of ways to baffle you for a few seconds I came up with:

Cryptogram A (*beginner*)
NGX?LXFA?DN?WQZZSTLIOH

Cryptogram B (*advanced*)
YYUCFPUM?GKIUEOS

If you haven't already submitted your solutions or figured out the solution and how it was done – or figured it out from the subtitle of this column; I used *monoalphabetic substitution* for cryptogram A and *polyalphabetic substitution* for cryptogram B. Lastly I purposely made sure that the cipher keys I used were independent enough from each other that breaking the first one (A) would not help you in breaking the second one (B). Though if you figured out how it was being done you would eventually figure them both out. Personally I love to figure stuff out so it was really hard holding back and waiting for this article to be released to the public. I haven't heard what the results of the submissions were at the time of this writing but I know we have quite a few crypto-connoisseurs onboard. Congratulations go all around for those that attempted to find the solution(s). To make the solutions easier to read I will break these cryptograms down separately from here on after we do a little 101.

***Alphabetic Substitution 101**

If you haven’t already guessed from the subtitle “***Quid Pro Quo***” substitution is all about *this* for *that*. Using the alphabet we are able to represent one letter for another. Not dissimilar to rotating letters (*see **Hacking Cryptograms 101**: Rotating letters is not secure, just annoying.)* [shifting to the left or write x many times] we are **substituting** one letter for another regardless of relative position.

Here is a very simple Monoalphabetic substitution example that applies the rotation of 1 to the cipher key. pA is substituted with cB, pB is substituted with cC and so on.

PLAINTEXT	HELLO WORLD
ALPHABET	ABCDEFGHIJKLMNOPQRSTUVWXYZ
CIPHERKEY	BCDEFGHIJKLMNOPQRSTUVWXYZA
CIPHERTEXT	IFMMP XPSME

It is a one for one substitution and can be easily solved without the use of a computer. You may also see these in your local newspaper’s puzzle section often defined as a cryptoquote or cryptogram. They are messages crypted via alphabetic substitution (most often monoalphabetic).

The example above can easily be solved using a simple Caesar brute force attack. Basically you are given there are 26 letters in the alphabet and the letters are being shifted to the left or the right. Thus it only takes a moment to generate all 26 possibilities thusly so (*generated with FTard Decoder Ring v2.0*):

0	IFMMP XPSME
1	**HELLO WORLD**
2	GDKKN VNQKC
3	FCJJM UMPJB
4	EBIIL TLOIA
5	DAHHK SKNHZ
6	CZGGJ RJMGY
7	BYFFI QILFX
8	AXEEH PHKEW
9	ZWDDG OGJDV
10	YVCCF NFICU
11	XUBBE MEHBT
12	WTAAD LDGAS
13	VSZZC KCFZR
14	URYYB JBEYQ
15	TQXXA IADXP
16	SPWWZ HZCWO
17	ROVVY GYBVN
18	QNUUX FXAUM
19	PMTTW EWZTL
20	OLSSV DVYSK
21	NKRRU CUXRJ
22	MJQQT BTWQI
23	LIPPS ASVPH
24	KHOOR ZRUOG
25	JGNNQ YQTNF
26	IFMMP XPSME

Immediately you will see that we find the solution through a brute force attack in the first rotation (highlighted in green). To make substitution more challenging to crack most often you will find they are not shifted because of this simple attack. Instead there is usually a pseudo-randomness applied where one letter is substituted by another that can be from any position along the alphabet. Such as pA being substituted by cQ. It is easier to keep track of the substitution process if the cipherkey used is a word that is known by parties meant to solve the cryptogram faster. Remember back in the day before the personal computer become a household object people were running around with paper and pencil (*imagine such a world now!*). Let's take our first example and modify it accordingly.

PLAINTEXT	HELLO WORLD
ALPHABET	ABCDEFGHIJKLMNOPQRSTUVWXYZ
CIPHERKEY	CRYPTOISFUNABDEGHJKLMQVWXZ
CIPHERTEXT	STAAE VEJAP

At first glance it may appear the ciphertext is using a variation of the Caesar cipher but when trying the same brute force attack as in the first example above all you get is more garbage without a solution (*at the level we are at*). The good thing we found out is that we aren't shifting to the left or right. With cryptanalysis it is always good to know what you aren't dealing with as well as what you are dealing with. With this specific ciphertext (STAAE VEJAP) we created simple cipherkey filtering out any repeating letters ("Crypto Is Fun") with the rest of the unused letters being appended at the end. Using this you can post the easy to remember cipherkey somewhere separate from the ciphertext itself. You are free to use any phrase as long as you stick to the idea that the phrase must be filtered of any repeating letters. For example if you use "**Mississippi**" as the prefix to your cipherkey you in reality can use "**Misp"** due to all the redundant letters in the phrase. The weakness of using an easy to remember cipherkey is that it may be easier to attack than a cipherkey that is "pseudo-random" or at least appears to be "pseudo-random". A cipherkey that uses a word in the dictionary can be attacked using a combination of dictionary and brute force methodology. For example such an attack could use the template of [word] + [alphabetical remainder] (e.g. "[ISRAEL] + [BCDFGHJKMNOPQTUVWXYZ] could be a possible cipherkey of "ISRAELBCDFGHJKMNOPQTUVWXYZ") and variations thereof. If a key isn't disposed of properly (such as never recycled or used again) it could prove value to the attacker in the future. The idea is not to become lazy and recycle keys otherwise once the key is found it can be reused again and again without bothering with dictionary or brute force attacking it first. The purpose of crypto is to slow down unintended parties from finding the original message in a timely fashion. By recycling the attacker only needs to find the key once and their efforts pay off because they don't need to invest more time into cracking it each time because they have the key already.

By the way you have just learned monoalphabetic substitution by using one (mono) alphabet to substitute another! The idea for polyalphabetic substitution is very similar except instead of using one alphabet you are using more than one. To keep it simple we will use the above examples with slight modification to fit our needs.

PLAINTEXT	HELLO WORLD
ALPHABET	ABCDEFGHIJKLMNOPQRSTUVWXYZ
CIPHERKEY	BCDEFGHIJKLMNOPQRSTUVWXYZA
	CDEFGHIJKLMNOPQRSTUVWXYZAB
CIPHERTEXT	IGMNP YPTMF

It may be ideal to use different alphabets but it isn’t required, you can use the same one but for visualization purposes we have made them slightly different by shifting the 2nd alphabet by one more. Now in this example for every letter we use we substitute with alternating the two alphabets. For example pA is substituted with cB, pB is substituted with cD. Confused? Let’s start the example again using the same plain letter. pA is substituted with cB (the upper alphabet), if we use pA again we use the lower alphabet which is cC, and if we use pA again we are back to cB, and so on. It is up to you if you are starting with the upper or lower alphabet or in the case of using more alphabets you can start it at whichever you like as long as you keep track of the alternation. Be aware the **key** as well as **the starting point** and **alternating process** make this more complex and thus are very important to remember otherwise the message will remain garbled when any of these three points are even just slightly off. (*Do not drink and crypt!*)

Using polyalphabetic substitution you will immediately notice we are no longer encoding with a 1:1 *unique* letter ratio. The same letter no longer represents any one letter of the plaintext. The advantage of this is that it is more difficult to use **frequency analysis** to figure out which letters may be more common than others in the English language (or whatever language is being used with this alphabet).

Solution A: Cryptogram A (*beginner*) NGX?LXFA?DN?WQZZSTLIOH

CIPHERTEXT	NGX?LXFA?DN?WQZZSTLIOH
ALPHABET	ABCDEFGHIJKLMNOPQRSTUVWXYZ
CIPHERKEY	QWERTYUIOPASDFGHJKLZXCVBNM
PLAINTEXT	YOU SUNK MY BATTLESHIP

This solution is pretty straightforward using monoalphabetic substitution. The question marks are obviously representation for spaces (they also deterred people from copy and pasting this string into some of the more common java crypto-decoders out there as most of them take the string literally and take longer to crack because they cannot immediately tell they should be ignored). The key was basic in the fact that it is your standard QWERTY layout. You guys using Dvorak should always keep this (common QWERTY keystrokes) in mind ;)

Solution B: Cryptogram B (*advanced*) YYUCFPUM?GKIUEOS

CIPHERTEXT	YYUCFPUM?GKIUEOS
ALPHABET	ABCDEFGHIJKLMNOPQRSTUVWXYZ
CIPHERKEY	MNBVCXZLKJHGFDSAPOIUYTREWQ
	POIUMNBVCXZYTREWQLKJHGFDSA
PLAINTEXT	ULTIMATE VICTORY

This solution used polyalphabetic substitution where the first alphabet (upper) is the reverse of the key in solution A (aka reverse QWERTY), and the second alphabet (lower) is a combination of reversal and displacement. (How is that for intermediate level?). This cryptogram was more complex also because the letter frequency was quite limited in words and spread.

Now let’s get to the fun part where I demonstrate how you can do this programmatically. This way if you run into this type of cryptogram again you can use this tool to quickly find the solution. If you aren’t into coding, I suggest you at least look it over to get a feel of the logic involved. If you simply refuse to even look at it skip the following example and **GOTO NO_CODING();**

Note: These are four separate documented examples that demonstrate monoalphabetic and polyalphabetic substitution as explained above when applied to Edition 3’s cryptograms (A and B) both for encoding and decoding. *Source is available for download in referenced links below.*

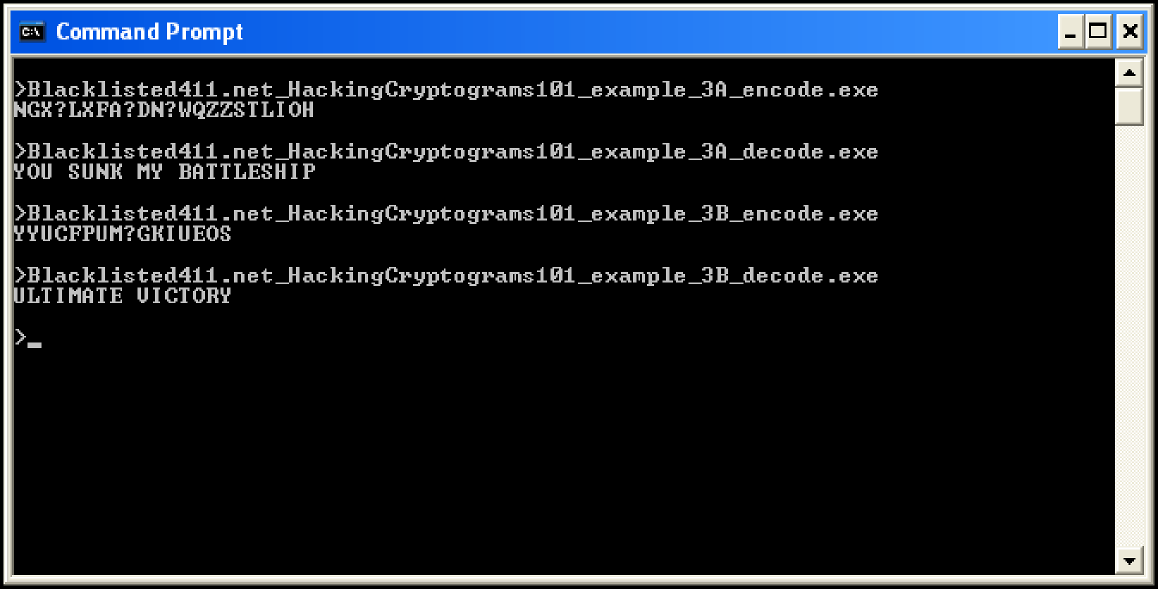
```
// Blacklisted411.net_HackingCryptograms101_example_3A.cpp
// Hacking Cryptograms 101: Quid Pro Quo
// By Israel Torres < israel@israeltorres.org >
//
// This example demonstrates how to use monoalphabetic substitution
// to encode plaintext into ciphertext using the cipherkey:
// QWERTYUIOPASDFGHJKLZXCVBNM
//

#include <iostream.h>
#include <string.h>

void main(int argc, char* argv[])
{
    // here we are setting up our variables for use
    char szCrypto[255]="\0";
    char cCTempA;

    // here we are copying our plaintext into a variable
    strcpy(szCrypto,"YOU SUNK MY BATTLESHIP");

    // here we are finding out what the length of the plaintext is
    int nCrypto = strlen(szCrypto);

    for (int nread = 0; nread < nCrypto; nread++)
    {
        cCTempA = szCrypto[nread];

        // In the following loop we are performing the substitution and outputting to the console

        // 65 - 90 CAPS ONLY
        switch (cCTempA)
        {
            case 'A': cout << "Q";
                break;
            case 'B': cout << "W";
                break;
            case 'C': cout << "E";
                break;
            case 'D': cout << "R";
                break;
            case 'E': cout << "T";
                break;
            case 'F': cout << "Y";
                break;
            case 'G': cout << "U";
                break;
            case 'H': cout << "I";
                break;
            case 'I': cout << "O";
                break;
            case 'J': cout << "P";
                break;
            case 'K': cout << "A";
                break;
            case 'L': cout << "S";
                break;
            case 'M': cout << "D";
                break;
            case 'N': cout << "F";
                break;
            case 'O': cout << "G";
                break;
            case 'P': cout << "H";
                break;
            case 'Q': cout << "J";
                break;
            case 'R': cout << "K";
                break;
                case 'S': cout << "L";
                    break;
                case 'T': cout << "Z";
                    break;
                case 'U': cout << "X";
                    break;
                case 'V': cout << "C";
                    break;
                case 'W': cout << "V";
                    break;
                case 'X': cout << "B";
                    break;
                case 'Y': cout << "N";
                    break;
                case 'Z': cout << "M";
                    break;

                default: cout << "?";  // space
            }

    }

    cout << endl;

}

// Blacklisted411.net_HackingCryptograms101_example_3A.cpp
// Hacking Cryptograms 101: Quid Pro Quo
// By Israel Torres < israel@israeltorres.org >
//
// This example demonstrates how to use monoalphabetic substitution
// to decode ciphertext into plaintext using the cipherkey:
// QWERTYUIOPASDFGHJKLZXCVBNM

#include <iostream.h>
#include <string.h>

void main(int argc, char* argv[])
{
    // here we are setting up our variables for use
    char szCrypto[255]="\0";
    char cCTempA;

    // here we are copying our cryptogram into a variable
    strcpy(szCrypto,"NGX?LXFA?DN?WQZZSTLIOH");

    // here we are finding out what the length of the cryptogram is
    int nCrypto = strlen(szCrypto);

    for (int nread = 0; nread < nCrypto; nread++)
    {
        cCTempA = szCrypto[nread];

        // In the following loop we are performing the substitution and outputting to the console

        // 65 - 90 CAPS ONLY
        switch (cCTempA)
        {
            case 'Q': cout << "A";
                break;
            case 'W': cout << "B";
                break;
            case 'E': cout << "C";
                break;
            case 'R': cout << "D";
                break;
            case 'T': cout << "E";
                break;
            case 'Y': cout << "F";
                break;
            case 'U': cout << "G";
                break;
            case 'I': cout << "H";
                break;
            case 'O': cout << "I";
```

```
                break;
            case 'P': cout << "J";
                break;
            case 'A': cout << "K";
                break;
            case 'S': cout << "L";
                break;
            case 'D': cout << "M";
                break;
            case 'F': cout << "N";
                break;
            case 'G': cout << "O";
                break;
            case 'H': cout << "P";
                break;
            case 'J': cout << "Q";
                break;
            case 'K': cout << "R";
                break;
            case 'L': cout << "S";
                break;
            case 'Z': cout << "T";
                break;
            case 'X': cout << "U";
                break;
            case 'C': cout << "V";
                break;
            case 'V': cout << "W";
                break;
            case 'B': cout << "X";
                break;
            case 'N': cout << "Y";
                break;
            case 'M': cout << "Z";
                break;

            default: cout << " ";  // space
        }

    }

    cout << endl;

}

// Blacklisted411.net_HackingCryptograms101_example_3A.cpp
// Hacking Cryptograms 101: Quid Pro Quo
// By Israel Torres < israel@israeltorres.org >
//
// This example demonstrates how to use polyalphabetic substitution
// to encode plaintext into ciphertext using the cipherkey:
// MNBVCXZLKJHGFDSAPOIUYTREWQ
// POIUMNBVCXZYTREWQLKJHGFDSA
//
#include <iostream.h>
#include <string.h>

void main(int argc, char* argv[])
{
    // here we are setting up our variables for use
    char szCrypto[255]="\0";
    char cCTempA;

    // here we are copying our plaintext into a variable
    strcpy(szCrypto,"ULTIMATE VICTORY");

    // here we are finding out what the length of the plaintext is
    int nCrypto = strlen(szCrypto);

    // for advanced we are using two alphabets (polyalphabetical) so the output will be two for 1
    for (int nread = 0; nread < nCrypto; nread++)
    {
        cCTempA = szCrypto[nread];

        // In the following loops we are performing the substitution and outputting to the console
// 65 - 90 CAPS ONLY
switch (cCTempA)
{
        case 'A': cout << "M";
                break;
        case 'B': cout << "N";
                break;
        case 'C': cout << "B";
                break;
        case 'D': cout << "V";
                break;
        case 'E': cout << "C";
                break;
        case 'F': cout << "X";
                break;
        case 'G': cout << "Z";
                break;
        case 'H': cout << "L";
                break;
        case 'I': cout << "K";
                break;
        case 'J': cout << "J";
                break;
        case 'K': cout << "H";
                break;
        case 'L': cout << "G";
                break;
        case 'M': cout << "F";
                break;
        case 'N': cout << "D";
                break;
        case 'O': cout << "S";
                break;
        case 'P': cout << "A";
                break;
        case 'Q': cout << "P";
                break;
        case 'R': cout << "O";
                break;
        case 'S': cout << "I";
                break;
        case 'T': cout << "U";
                break;
        case 'U': cout << "Y";
                break;
        case 'V': cout << "T";
                break;
        case 'W': cout << "R";
                break;
        case 'X': cout << "E";
                break;
        case 'Y': cout << "W";
                break;
        case 'Z': cout << "Q";
                break;

        default: cout << "?";  // space
}

nread++;
cCTempA = szCrypto[nread];

// 65 - 90 CAPS ONLY // 2nd time around
switch (cCTempA)
{
        case 'A': cout << "P";
                break;
        case 'B': cout << "O";
                break;
        case 'C': cout << "I";
                break;
        case 'D': cout << "U";
                break;
        case 'E': cout << "M";
```

```
                break;
            case 'F': cout << "N";
                break;
            case 'G': cout << "B";
                break;
            case 'H': cout << "V";
                break;
            case 'I': cout << "C";
                break;
            case 'J': cout << "X";
                break;
            case 'K': cout << "Z";
                break;
            case 'L': cout << "Y";
                break;
            case 'M': cout << "T";
                break;
            case 'N': cout << "R";
                break;
            case 'O': cout << "E";
                break;
            case 'P': cout << "W";
                break;
            case 'Q': cout << "Q";
                break;
            case 'R': cout << "L";
                break;
            case 'S': cout << "K";
                break;
            case 'T': cout << "J";
                break;
            case 'U': cout << "H";
                break;
            case 'V': cout << "G";
                break;
            case 'W': cout << "F";
                break;
            case 'X': cout << "D";
                break;
            case 'Y': cout << "S";
                break;
            case 'Z': cout << "A";
                break;

            default: cout << "!";  // space
        }

    }

    cout << endl;

}

// Blacklisted411.net_HackingCryptograms101_example_3A.cpp
// Hacking Cryptograms 101: Quid Pro Quo
// By Israel Torres < israel@israeltorres.org >
//
// This example demonstrates how to use polyalphabetic substitution
// to encode plaintext into ciphertext using the cipherkey:
// MNBVCXZLKJHGFDSAPOIUYTREWQ
// POIUMNBVCXZYTREWQLKJHGFDSA
//

#include <iostream.h>
#include <string.h>

void main(int argc, char* argv[])
{
    // here we are setting up our variables for use
    char szCrypto[255]="\0";
    char cCTempA;

    // here we are copying our cryptogram into a variable
    strcpy(szCrypto,"YYUCFPUM?GKIUEOS");
```

```
// here we are finding out what the length of the cryptogram is
int nCrypto = strlen(szCrypto);

// for advanced we are using two alphabets (polyalphabetical) so the output will be two for 1
for (int nread = 0; nread < nCrypto; nread++)
{
        cCTempA = szCrypto[nread];

        // In the following loops we are performing the substitution and outputting to the console

        // 65 - 90 CAPS ONLY
        switch (cCTempA)
        {
                case 'M': cout << "A";
                        break;
                case 'N': cout << "B";
                        break;
                case 'B': cout << "C";
                        break;
                case 'V': cout << "D";
                        break;
                case 'C': cout << "E";
                        break;
                case 'X': cout << "F";
                        break;
                case 'Z': cout << "G";
                        break;
                case 'L': cout << "H";
                        break;
                case 'K': cout << "I";
                        break;
                case 'J': cout << "J";
                        break;
                case 'H': cout << "K";
                        break;
                case 'G': cout << "K";
                        break;
                case 'F': cout << "M";
                        break;
                case 'D': cout << "N";
                        break;
                case 'S': cout << "O";
                        break;
                case 'A': cout << "P";
                        break;
                case 'P': cout << "Q";
                        break;
                case 'O': cout << "R";
                        break;
                case 'I': cout << "S";
                        break;
                case 'U': cout << "T";
                        break;
                case 'Y': cout << "U";
                        break;
                case 'T': cout << "V";
                        break;
                case 'R': cout << "W";
                        break;
                case 'E': cout << "X";
                        break;
                case 'W': cout << "Y";
                        break;
                case 'Q': cout << "Z";
                        break;

                default: cout << " ";  // space
        }

        nread++;
        cCTempA = szCrypto[nread];

        // 65 - 90 CAPS ONLY // 2nd time around
        switch (cCTempA)
        {
            case 'P': cout << "A";
                break;
            case 'O': cout << "B";
                break;
            case 'I': cout << "C";
                break;
            case 'U': cout << "D";
                break;
            case 'M': cout << "E";
                break;
            case 'N': cout << "F";
                break;
            case 'B': cout << "G";
                break;
            case 'V': cout << "H";
                break;
            case 'C': cout << "I";
                break;
            case 'X': cout << "J";
                break;
            case 'Z': cout << "K";
                break;
            case 'Y': cout << "L";
                break;
            case 'T': cout << "M";
                break;
            case 'R': cout << "N";
                break;
            case 'E': cout << "O";
                break;
            case 'W': cout << "P";
                break;
            case 'Q': cout << "Q";
                break;
            case 'L': cout << "R";
                break;
            case 'K': cout << "S";
                break;
            case 'J': cout << "T";
                break;
            case 'H': cout << "U";
                break;
            case 'G': cout << "V";
                break;
            case 'F': cout << "W";
                break;
            case 'D': cout << "X";
                break;
            case 'S': cout << "Y";
                break;
            case 'A': cout << "Z";
                break;

            default: cout << " ";  // space
        }

    }

    cout << endl;

}
```

Here is the screenshot of the 4 (A&B | encode&decode) compiled programs:



```
>Blacklisted411.net_HackingCryptograms101_example_3A_encode.exe
NGX?LXFA?DN?WQZZSTLIOH

>Blacklisted411.net_HackingCryptograms101_example_3A_decode.exe
YOU SUNK MY BATTLESHIP

>Blacklisted411.net_HackingCryptograms101_example_3B_encode.exe
YYUCFPUM?GKIUEOS

>Blacklisted411.net_HackingCryptograms101_example_3B_decode.exe
ULTIMATE VICTORY

>_
```

NO_CODING()

*Alphabetic (regardless of mono or poly) substitution is better to use than rotating letters because it makes it slightly more difficult to attack in a short amount of time. There are plenty of automatic solvers out there that may find solutions to your substituted ciphertext (mostly for newspaper cryptoquotes/cryptograms). Start by searching google for “crypto solver” to find them. You’ll quickly learn that most are geared for monoalphabetic ciphers and are useless when attempting to solve polyalphabetic substitutions. One of my crypto-related projects include involving better solvers that don’t take an eternity to find a solution – it is on the backburner, but will be out with time.

This concludes Lesson 3 of ***Hacking Cryptograms 101***. Hopefully you enjoy crypto as much as I do. See you next edition!

Keeping it ‘rael,
Israel Torres

Source Code

Source code can be downloaded here:
Bundled (zip):
http://blacklisted411.israeltorres.org/Blacklisted411.net_HackingCryptograms101_example_3_all.zip
individual (cpp):
http://blacklisted411.israeltorres.org/Blacklisted411.net_HackingCryptograms101_example_3A_encode.cpp
http://blacklisted411.israeltorres.org/Blacklisted411.net_HackingCryptograms101_example_3A_decode.cpp
http://blacklisted411.israeltorres.org/Blacklisted411.net_HackingCryptograms101_example_3B_encode.cpp
http://blacklisted411.israeltorres.org/Blacklisted411.net_HackingCryptograms101_example_3B_decode.cpp

MD5 hash of the source code can be validated at:
http://blacklisted411.israeltorres.org/Blacklisted411.net_HackingCryptograms101_example_3_all.zip.md5.txt
http://blacklisted411.israeltorres.org/Blacklisted411.net_HackingCryptograms101_example_3_all.cpp.md5.txt

note: you may have to copy and paste the addresses above if they wrap around!

References and Related Links

Cryptography
http://en.wikipedia.org/wiki/Cryptography

Monoalphabetic Substitution Cipher
http://en.wikipedia.org/wiki/Monoalphabetic_substitution_cipher

Polyalphabetic Substitution Cipher
http://en.wikipedia.org/wiki/Polyalphabetic_cipher

FTard Decoder Ring v2.0
http://tools.israeltorres.org/#FTardDecoderRing

Quid pro quo
http://en.wikipedia.org/wiki/Quid_pro_quo

[b]

TEAM ICE & THE ELUSIVE XBOX 360 MOD CHIP

Blacklisted411

PUBLIC ADVISORY
BY USTLER OF BLACKLISTED 411 MAGAZINE

What started as an innocent curiosity into the Xbox 360 modchip revealed a lot of information that the public was not aware of. My original intention was to do an interview with Team ICE, who we honestly believed to have had a legitimate product. Through our communications, we soon became aware of information which that gave us serious doubts into their claims. Not wanting to publish false information, we made an effort to investigate every aspect of Team ICE and their past. The following information is provided in order to help you make your OWN decision on the matter. We hope that through verifiable information (some more than others), you will be able to draw a reasonable conclusion into the validity of their claims.

Note to Team ICE: Our original intentions were not tainted, but through our communication and investigation, Blacklisted411 cannot endorse the product without verifiable proof. If you do have a legitimate product, we do apologize (although my personal conclusion does not agree). Your lack of co-operation into providing detail, and the blatantly wrong information you provided us has brought this upon you.

Rarecodes.com – icemodchip.com – infinitymods.com

Deception or mistake?
First, I would like to point out that infitymods.com have no apparent relation to infinitymod.com (the makers of the Matrix Infinity modchip for PS2). Furthermore, the name icemodchip also has no relation to the “Xenium Ice” modchip produced by Team Xodus. Many people have concluded that the project was lead by Team Xodus, even though this is not the case.

Is there a relationship between the sites?

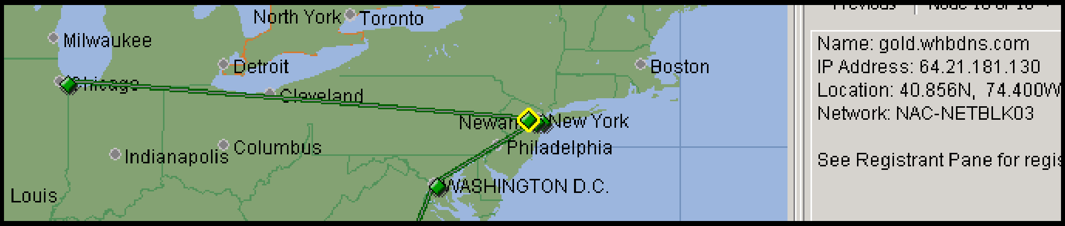
```
X-PopBeforeSMTPSenders:
affils@icemodchip.com,ebay@infinitymods.com,evan@infinitymods.com,faq@icemodchip.co
info@icemodchip.com,installers@icemodchip.com,orders@infinitymods.com,preorders@icem
resellers@icemodchip.com,sales@infinitymods.com,sales@rarecodes.com,
support@infinitymods.com,support@rarecodes.com,teamlead@icemodchip.com,
wholesale@icemodchip.com
```

The next claim started circulating not long after my initial contact with Team ICE had started. Upon further investigation, I found that rarecodes.com, icemodchip.com and infinitymods.com had the same IP address (odd yet interesting). A simple ping gave us the IP address of 64.21.181.130 on all three URLs. Directly accessing the IP address gives us a Cpanel message informing us that the IP address has no default host (meaning that the websites are all virtually hosted). One could easily claim that the web host, rarecodes.com, was simply virtually hosting their customers on a single IP address. The next piece of information retrieved from the header of an email sent by Team ICE shows the danger in not configuring your SMTP server properly.

This leak of information provides us with a complete list of email addresses being used on the server. Notice that only icemodchip.com, infinitymods.com and rarecodes.com appear. It's rather odd that a hosting company such as rarecodes.com would allow such a leak of information. Simply by using nslookup we were able to do a Reverse IP address lookup (meaning that instead of resolving a domain to an IP address, we resolve an IP address to a domain). Using this, we got the hostname gold.whbdns.com. Next, we wanted to validate the claim that rarecodes.com makes on their website.

*"**RareCodes Inc.** is located in NL, Canada. We currently maintain 10 main servers locally, and 32 mirror servers located in various U.S. States. All 42 of our servers are running into a direct GiGaBiT Pipeline, which guarantees you the customer fast speeds, at all times"*

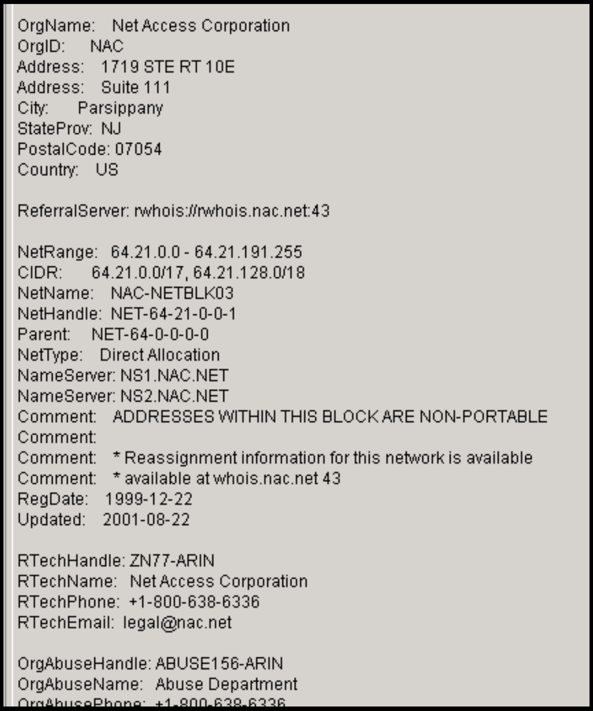
42 servers and ten of them are in Canada? Well, using Neotrace (Similar to VisualRoute) we were able to create a quick map showing where the server is located. For obvious reasons, the trace route was routed through a proxy and the portion containing the origination of the trace was omitted.



Name: gold.whbdns.com
IP Address: 64.21.181.130
Location: 40.856N, 74.400W
Network: NAC-NETBLK03

See Registrant Pane for regis

Furthermore, we determined that the IP address is assigned to Net Access Corporation (Although we are pretty certain they are only the host and have no involvement in icemodchips.com or any of the other websites in question.)



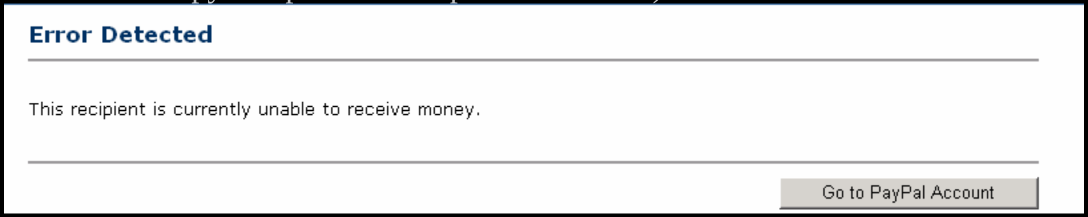
```
OrgName:    Net Access Corporation
OrgID:      NAC
Address:    1719 STE RT 10E
Address:    Suite 111
City:       Parsippany
StateProv:  NJ
PostalCode: 07054
Country:    US

ReferralServer: rwhois://rwhois.nac.net:43

NetRange:   64.21.0.0 - 64.21.191.255
CIDR:       64.21.0.0/17, 64.21.128.0/18
NetName:    NAC-NETBLK03
NetHandle:  NET-64-21-0-0-1
Parent:     NET-64-0-0-0-0
NetType:    Direct Allocation
NameServer: NS1.NAC.NET
NameServer: NS2.NAC.NET
Comment:    ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE
Comment:
Comment:    * Reassignment information for this network is available
Comment:    * available at whois.nac.net 43
RegDate:    1999-12-22
Updated:    2001-08-22

RTechHandle: ZN77-ARIN
RTechName:   Net Access Corporation
RTechPhone:  +1-800-638-6336
RTechEmail:  legal@nac.net

OrgAbuseHandle: ABUSE156-ARIN
OrgAbuseName:   Abuse Department
OrgAbusePhone:  +1-800-638-6336
```

In order to verify that rarecodes.com had no involvement, we attempted to contact them by phone. Much to our disappointment, the phone number listed on their website "709-749-0373" was not working. Instead of being greeted by Rarecodes, we got this strange message:

Roger's Wireless: The cellular number you have called is not assigned, please check the number and try the call again. (Then repeated in French)

What happened to the 24 hour tech support? Why would a company like rarecodes.com with their 48 servers and 10 local servers choose to associate themselves with a modchip website??

Of course, we were not done yet. A few pieces of additional information prove to be somewhat incriminating. First off, rarecodes.com uses paypal. The email address used for the paypal account is support@rarecodes.com, but using this link to pay him gives us this (Note the link is below but in small font. Feel free to copy and paste it to duplicate the error):



Error Detected

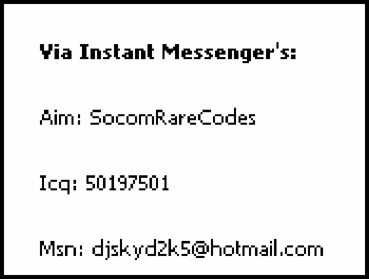
This recipient is currently unable to receive money.

https://www.paypal.com/cgi-bin/webscr?cmd=_xclick-subscriptions&business=support@rarecodes.com&item_name=Basic%20Plan%20-%201%20Month&item_number=RC-0049-BAC1MT&no_shipping=1&no_note=1&currency_code=USD&bn=PP-SubscriptionsBF&charset=UTF-8&a1=0.00&p1=3&t1=D&a3=9.25&p3=1&t3=M&src=1&sra=1

This obviously raises the question on how rarecodes.com receives payment? (icemodchip.com must have some way of paying them? Why haven't they fixed this? With so many customers (25,000) this would seem an important issue to fix, and quickly.

At this point, I was pretty confident that rarecodes.com was indeed tied to icemodchip.com on more than a business level. In fact, my personal belief is that they are indeed the same entity (although this is only my personal view).

The next piece of information was pretty funny. It's contained on the "Contact Us" section of rarecodes.com and is shown below:



Via Instant Messenger's:

Aim: SocomRareCodes

Icq: 50197501

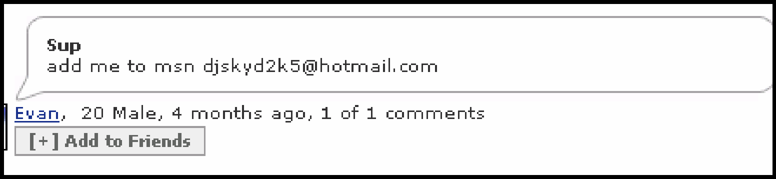
Msn: djskyd2k5@hotmail.com

http://rarecodes.com/aboutus.php

SkyD – Tracking and his past.

Information is like a puzzle piece. The more information you have on hand, the better your look of the whole picture. The information we are talking about here is the MSN email address. Using this in Google, we obtained a rather interesting site called sms.ac. The following links provide you with a look at SkyD in his true form (If you don't think SkyD is in fact running icemodchip.com, keep reading, we have more stuff to show you)

> IMPORTANT: An email correspondence between Blacklisted411 and icemodchip.com confirms the person behind this is SkyD. When asked if he used SkyD as his alias in the past, he confirmed it by saying "Quite possibly :P". We take this to be a positive link between SkyD and icemodchip.com. If he wasn't SkyD, we doubt he would even allude to the possibility. Even without this information, evidence still provides a decent connection between the two.



Sup
add me to msn djskyd2k5@hotmail.com

Evan, 20 Male, 4 months ago, 1 of 1 comments

A photo album of SkyD is available at
http://sms.ac/MPA/Default.aspx?u=DJSkyD

A description of himself
http://sms.ac/AboutMe/AboutMe_View.aspx?u=DJSkyD

In fact, if you look through his buddy list, you may find someone that appeared in the initial video that icemodchip.com released.

Notice one important thing. "4 Months ago".. This guy is only 20 years old! By the way, off topic **sms.ac allows you to send our modchip friend a SMS message. Of course we don't endorse this.**

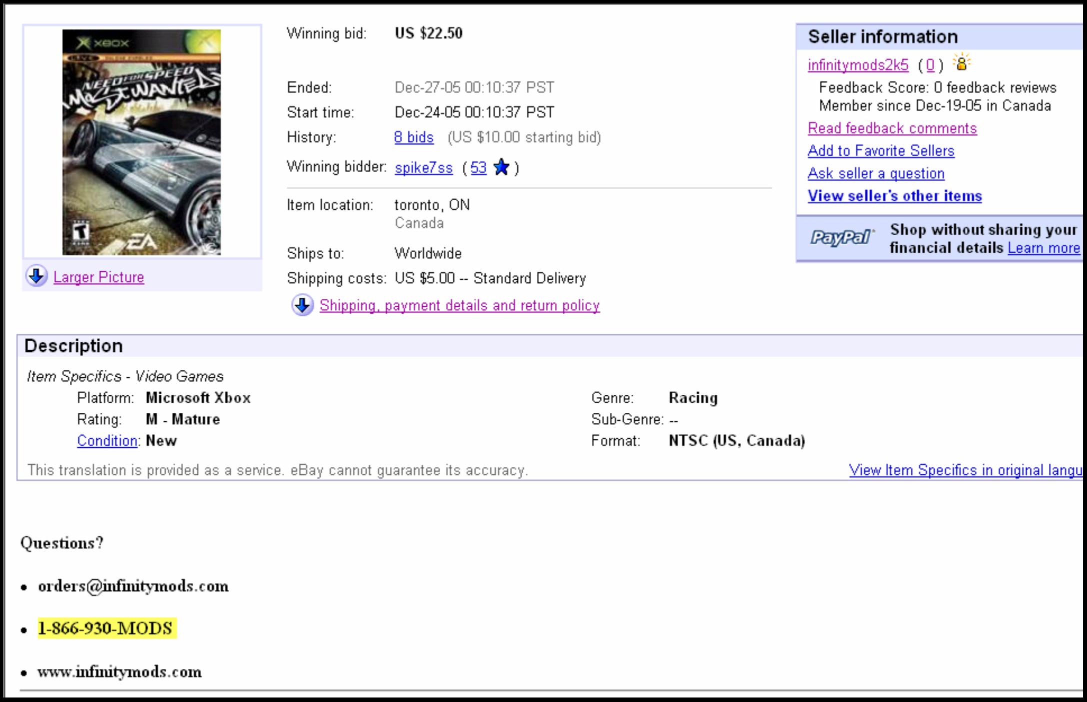
Important: gamesfirst.com provides this information for us:
“Evan OKeefe (we assume O'Keefe) is listed as the admin, billing, and technical contact for icemodchip.com. According to the Contact Us information on InfinityMods.com, evan[AT]infinitymods.com will also put you in contact with the InfinityMods sales department. The first assumption is that evan@infinitymods is Evan O'Keefe, but that could be wrong. Team I.C.E. has repeatedly denied that InfinityMods.com and Team I.C.E. are the same, saying that InfinityMods is an affiliate.”
http://gamesfirst.com/?id=1141

Now, I told you guys to remember his name. One important piece of information was released before which allowed us to make a connection between the two people. If you remember, the X-PopBeforeSMTPSenders statement that we obtained from an email sent had the name Evan@infinitymods.com. What a coincidence.

Ebay Fraud?
Another thing we noticed in the X-PopBeforeSMTPSenders that was in the header of the email address was Ebay@infinitymods.com. This brought us to the conclusion that infinitymods.com may be selling on eBay. A quick look at infinitymods.com gave us some rather useful information. Since we are unable to lookup a UserID on Ebay using an email address, we must use other useful information to match a description. In this case, all we needed was the phone number (1-866-930-mods). Using this, we got this auction:

Infinity-Mods
1-866-930-MODS

Need For Speed Most Wanted *Christmas Special PS2*
http://64.233.187.104/search?q=cache:PzfIqyV_WmQJ:cgi.ebay.com/Need-For-Speed-Most-Wanted-Christmas-Special-PS2_W0QQitemZ8246000967QQcategoryZ62053QQcmdZViewItem+1-866-930-mods&hl=en
(Text is made small to save space. Its still a working link but we were required to use Google cache to obtain the original auction since eBay removed it.)



Winning bid: **US $22.50**

Ended: Dec-27-05 00:10:37 PST
Start time: Dec-24-05 00:10:37 PST
History: 8 bids (US $10.00 starting bid)
Winning bidder: spike7ss (53 ★)

Item location: toronto, ON
Canada

Ships to: Worldwide
Shipping costs: US $5.00 -- Standard Delivery
Shipping, payment details and return policy

Seller information

infinitymods2k5 (0)
Feedback Score: 0 feedback reviews
Member since Dec-19-05 in Canada
Read feedback comments
Add to Favorite Sellers
Ask seller a question
View seller's other items

PayPal **Shop without sharing your financial details** Learn more

Description

Item Specifics - Video Games

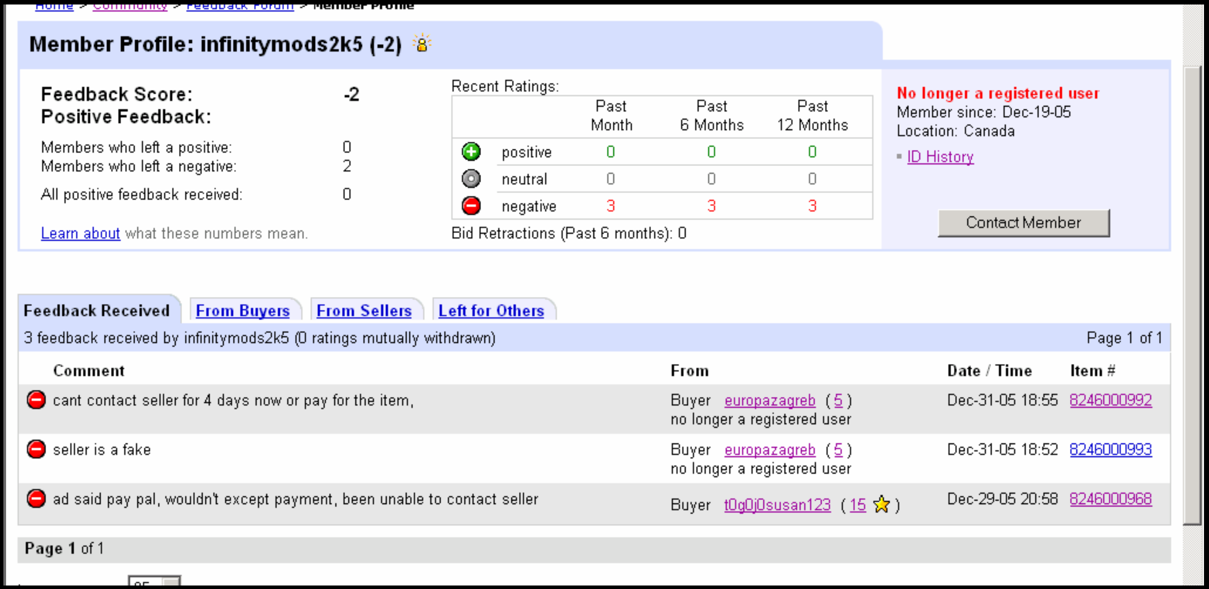
Platform:	**Microsoft Xbox**	Genre:	**Racing**
Rating:	**M - Mature**	Sub-Genre:	--
Condition:	**New**	Format:	**NTSC (US, Canada)**

This translation is provided as a service. eBay cannot guarantee its accuracy. View Item Specifics in original langu

Questions?

- orders@infinitymods.com
- 1-866-930-MODS
- www.infinitymods.com

This auction started on the 12-24-05 and ended on 12-27-05. The good part about it was that we were able to pick up his userID: **infinitymods2k5** (Registered on December 19th 2005). A quick look at his feedback gave us this information:



Home > Community > Feedback Forum > **Member Profile**

Member Profile: infinitymods2k5 (-2)

Feedback Score:	-2
Positive Feedback:	
Members who left a positive:	0
Members who left a negative:	2
All positive feedback received:	0

Learn about what these numbers mean.

Recent Ratings:

	Past Month	Past 6 Months	Past 12 Months
positive	0	0	0
neutral	0	0	0
negative	3	3	3

Bid Retractions (Past 6 months): 0

No longer a registered user
Member since: Dec-19-05
Location: Canada
ID History

3 feedback received by infinitymods2k5 (0 ratings mutually withdrawn) Page 1 of 1

Comment	From	Date / Time	Item #
cant contact seller for 4 days now or pay for the item,	Buyer europazagreb (5) no longer a registered user	Dec-31-05 18:55	8246000992
seller is a fake	Buyer europazagreb (5) no longer a registered user	Dec-31-05 18:52	8246000993
ad said pay pal, wouldn't except payment, been unable to contact seller	Buyer t0g0j0susan123 (15 ★)	Dec-29-05 20:58	8246000968

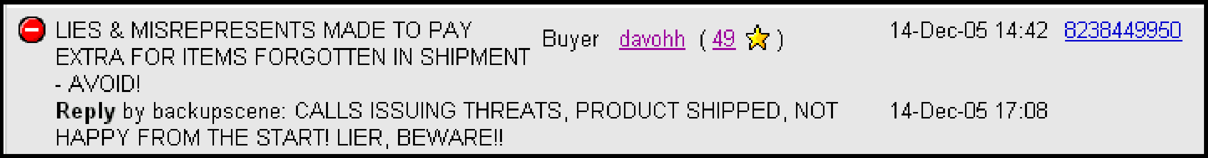
Page 1 of 1

http://feedback.ebay.com/ws/eBayISAPI.dll?ViewFeedback&userid=infinitymods2k5

Three negative feedback in less than 20 days. Also, one might question the integrity of the buyer in this case (since some buyers often never pay for an item.) If you notice the ending times for europazagreb (one of the ebay buyers) are about the same, so we doubt that he was attempting to defraud infinitymods2k5.

We also found another eBay account under the name “backupscene” by using the email from rarecodes.com. Although there are a few are some positive feedback left for him, we would like to point out this high price item http://cgi.ebay.ca/XBOX-360-ULTIMATE-PACKAGE_W0QQitemZ8238449950QQcategoryZ62054QQtcZphotoQQrdZ1QQcmdZViewItem

This feedback was left on Dec 14th.



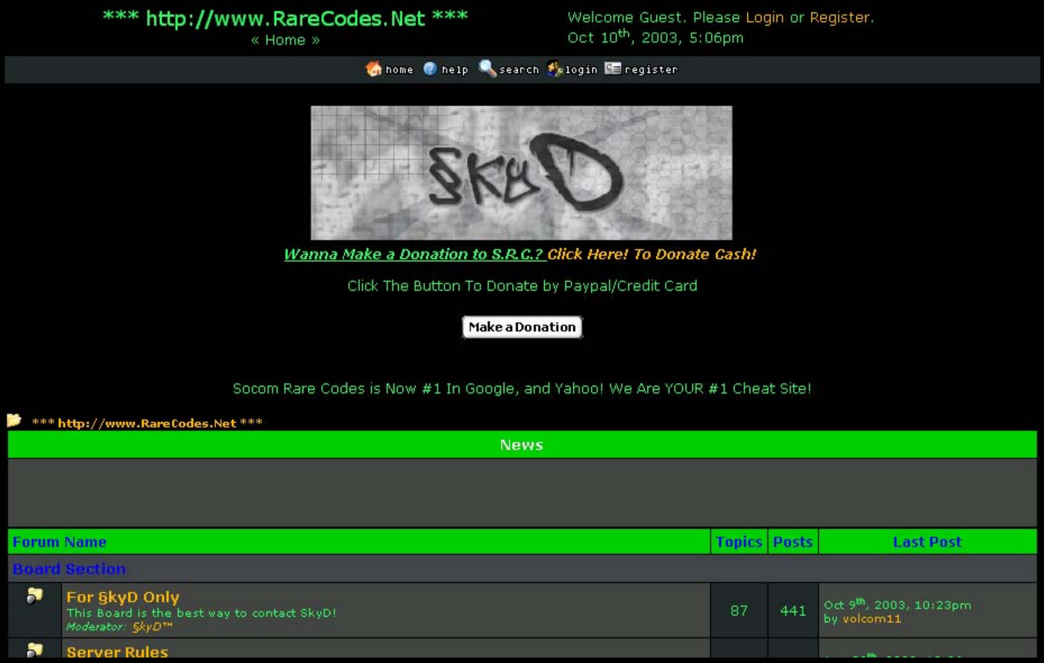
⛔ LIES & MISREPRESENTS MADE TO PAY EXTRA FOR ITEMS FORGOTTEN IN SHIPMENT - AVOID!	Buyer davohh (49 ☆)	14-Dec-05 14:42	8238449950
Reply by backupscene: CALLS ISSUING THREATS, PRODUCT SHIPPED, NOT HAPPY FROM THE START! LIER, BEWARE!!		14-Dec-05 17:08	

Most notably, this is a high priced item (942$ USD Aprox). In a previous .NET edition, I talked about how eBay fraud is committed. Starting with small items being bought and sold, the fraudulent eBay seller will eventually attempt to sell a big ticket item. Defrauding people of 20 USD is not worth it, but 900+ USD may be worth the 2 months of selling profitless items. Please note, backupscene is no longer a registered eBay member. The registration date was Nov 24, 2005 and the last sale was around the 19th of December.

The past of rarecodes.com

I want to point out some interesting facts about rarecodes.com past. Using one of my favorite websites (archive.org) we are able to reconstruct every change made over the past 3 years.

The original rarecodes.com site was simply a redirect to a hyperboards site and was used for cheat codes (More like bugs) in Socom for PS2. This page appears on Oct 17, 2003
(Preview created with archive.org)



*** http://www.RareCodes.Net ***
« Home »
Welcome Guest. Please Login or Register.
home
help
search
login
register
§kyD
Make a Donation

Oct 10th, 2003, 5:06pm

Wanna Make a Donation to S.R.C.? Click Here! To Donate Cash!

Click The Button To Donate by Paypal/Credit Card

Socom Rare Codes is Now #1 In Google, and Yahoo! We Are YOUR #1 Cheat Site!

*** http://www.RareCodes.Net ***

News

Forum Name	Topics	Posts	Last Post
Board Section			
For §kyD Only This Board is the best way to contact SkyD! *Moderator: §kyD™*	87	441	Oct 9th, 2003, 10:23pm by volcom11
Server Rules			

http://web.archive.org/web/20031017083434/http://www.rarecodes.com/

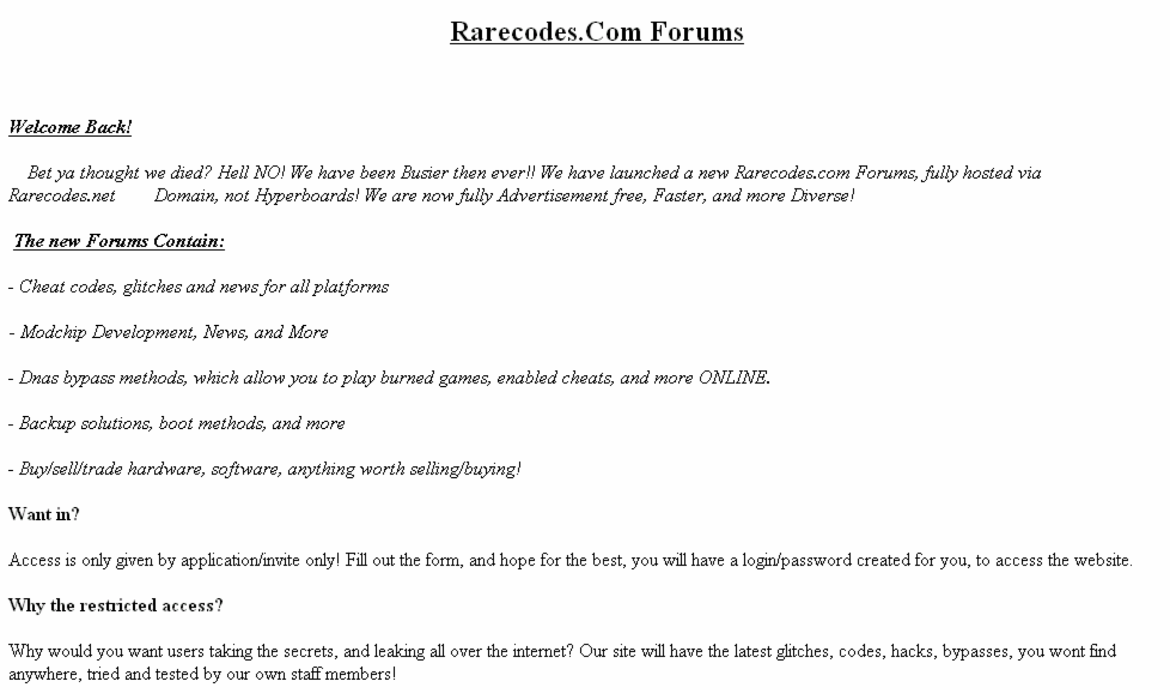
On Dec 27, 2003, the site made a drastic change to this (Preview not included).

http://web.archive.org/web/20031227022030/http://www.rarecodes.com/

In 2004, it changed to what appears to be a PHP based auction site, but to my knowledge, this auction site was never fully functional. The original programming claims to be SkyD, but I have been unable to confirm this (although I remember seeing a PHP classifieds script that look very similar about 2 years ago).

http://web.archive.org/web/20040203201458/http://www.rarecodes.com/

Not long after the classified script disappeared, this popped up.



Rarecodes.Com Forums

Welcome Back!

Bet ya thought we died? Hell NO! We have been Busier then ever!! We have launched a new Rarecodes.com Forums, fully hosted via Rarecodes.net Domain, not Hyperboards! We are now fully Advertisement free, Faster, and more Diverse!

The new Forums Contain:

- Cheat codes, glitches and news for all platforms

- Modchip Development, News, and More

- Dnas bypass methods, which allow you to play burned games, enabled cheats, and more ONLINE.

- Backup solutions, boot methods, and more

- Buy/sell/trade hardware, software, anything worth selling/buying!

Want in?

Access is only given by application/invite only! Fill out the form, and hope for the best, you will have a login/password created for you, to access the website.

Why the restricted access?

Why would you want users taking the secrets, and leaking all over the internet? Our site will have the latest glitches, codes, hacks, bypasses, you wont find anywhere, tried and tested by our own staff members!

http://web.archive.org/web/20040609015146/http://www.rarecodes.net/

After this was posted for some time, he appeared to have changed webhosts to mixcat.net and created a “modchip” website. Here he took “Pre-orders” but information leading to customers is very vague.

http://web.archive.org/web/20040624063118/http://www.rarecodes.com/

We did however find this

> *”This guy is a scammer, he owes me $50 for modding someone's PS2, he screwed me out of money and I demand it, I have waited over 2 months and the time has come to put a stop to the scams.* ***SkyD*** *if you...”*

http://64.233.187.104/search?q=cache:4zUBgs54JkIJ:www.backup-source.com/forums/search.php%3Fdo%3Dfinduser%26u%3D2925+rarecodes.com+skyd&hl=en

A lot more information is out there on his previous history. Try looking for SkyD and rarecodes.com in google (Combined). Remember this was in 2004, so use google cache if it’s old and outdated. What you will find is that SkyD got himself banned from a myriad of forums. Most of the people I could find had posted something similar to the above quote. Contacting these people became extremely challenging. Those who did return our emails expressed the same views but did not want to be mentioned in this release due to the fact they no longer cared about the situation. That pretty much covers the history of rarecodes.com. I do wonder where rarecodes.com 48 servers and 4 years of service are?

Good templates don't make a website!!!

Next topic we cover is the use of templates. This section will be rather short. Both rarecodes.com and icemodchip.com use templates that can easily be found on P2P. The templates originate from boxedart.com and a simple membership allows you to download both of them for a low price.

Icemodchip.com = http://www.boxedart.com/phpshop/info.php/mp_/16964
Rarecodes.com = http://www.boxedart.com/phpshop/info.php/mp_/16145

It is a strange coincidence that both templates come from the same site, but this isn't enough evidence in itself. Everything is pretty much the same, even down to the annoying techno music. I've seen a few claims from rarecodes.com on various message boards (Under the name SkyD) telling people they have a team of people designing stuff, but this doesn't plausible.

SkyD and icemodchip.com

Some of you may be doubt that SkyD has any involvement in icemodchip.com. I would like to direct you back a few years. A project called "Project Lithium 2" (I want to know where Project Lithium 1 went) was in "Development" by a person named SkyD. According to sources (please remember this is 2004, so Google cache becomes your best friend again. Also a few websites still have the info.)

> *"We received a brief status update on **Project Lithium 2** from the coder **SkyD** over the weekend listing **June 15th, 2004** as the official BETA test date. It's important to NOTE that the final (public) version of **Project Lithium 2** (tentatively scheduled for a July 15th release) will include a pressed factory-produced PS2 disc expected to **cost $19.95** each."*

http://64.233.187.104/search?q=cache:ij5tCbGQU4MJ:www.ps2ownz.com/modules.php%3Fname%3DNews%26file%3Dshowarticle%26threadid%3D20428+%22Project+Lithium+2%22+cost&hl=en

The problem was that I was unable to find ANY information about it ever being released. I did, however, find this on the same site as the above quote (This was released before the 19.95 price quote).

> *"Based off of **NoID**'s original work and the past NIP files, **Project Lithium 2** is currently in the final development/packaging process and is coded by **SkyD**."*

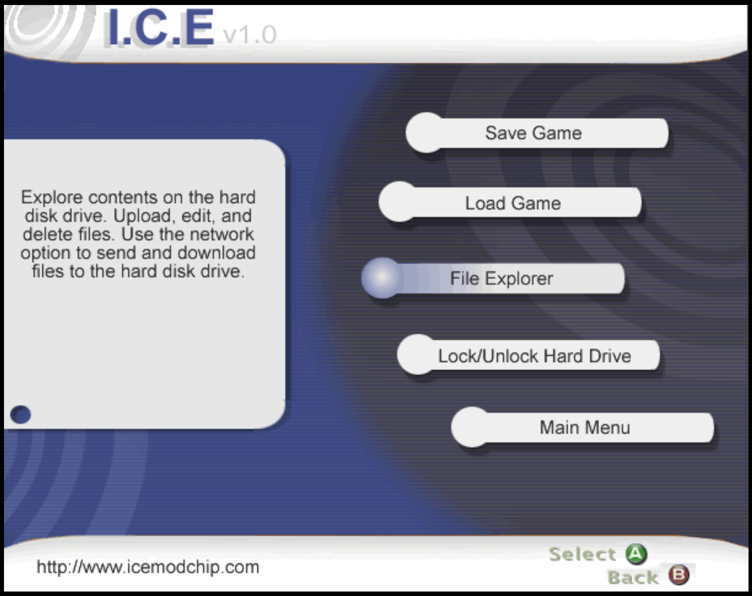
http://64.233.187.104/search?q=cache:VF_0dl46_0QJ:www.ps2ownz.com/modules.php%3Fname%3DNews%26file%3Dshowarticle%26t%3D20026%26op%3Dnextnewest+%22Project+Lithium+2%22+&hl=en

Notice it says "Based off NoID's original work." This makes us wonder if the project was simply NoID's work on a CD with newer NIP files. No apparent production was ever made and the project seemed to disappear after awhile.

Problems in Menu

Another thing we noted while doing research was the recently released Menu items are completely wrong. The Xbox 360 does not have an internal hard drive, but rather a removable SATA drive. Sources inside Xbox-Scene tell us that the Xbox 360 hard drive does not employ locking techniques found in the original

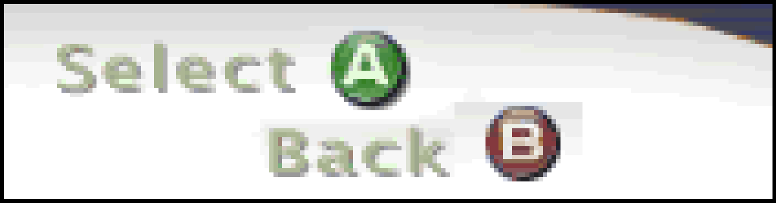
Xbox. Furthermore some simple research on Google shows us multiple sources confirming that the Xbox 360 hard drive is indeed NOT locked. So this brings up the question, why is this option in the menu?



Explore contents on the hard disk drive. Upload, edit, and delete files. Use the network option to send and download files to the hard disk drive.

As we can see, the “Lock/Unlock Hard Drive” is fake since the hard drive is not locked by the Xbox 360 console. Also notice the “Save Game” and “Load Game” menu items. Someone please explain to me why these are in a bios menu??? Isn’t the game responsible for saving and loading game progress??

Furthermore, a quick look at the detail shows us this.



Look right above the B button. Notice the background around it. This is an obvious Photoshop copy and paste. While the menu does look semi-authentic, it is entirely possible that it may be a modified version of a previous dashboard skin or bios skin from the original Xbox. We didn't want to spend time looking for the skin. When we contacted icemodchip.com about these inaccuracies, they responded by removing the images from their site and telling us that they were ONLY concept images and that the menu in the modchip was different.

Inaccuracies in email claims

The following is email correspondence between me and Team I.C.E.
In response to a question asked about who had mod chips for testing, we got this reply.

> Xbox-scene has one, foundmy.com has 1, and hack360.com has one. More samples to be sent shortly.

We attempted to contact Xbox-Scene and Hack360. Xbox-Scene replied and confirmed our suspicions that they had indeed not received a single thing. Currently, we have still not heard back from Hack360 (But I personally believe this is true of all three of the mentioned sites). After contacting him back asking why they denied having received the modchip from him, he told us this:

> People with samples have a close lid on this matter as there is a security issue at hand

I would like to note that there is no security matter. All I asked was whether this claim was true. The response, Absolutely NOT. In fact, when posting the question on the forum, I was greeted with a rude reply and the topic was locked and deleted (Supposedly someone was creating multiple names and posting links to icemodchip.com along with "false" information [Or that's at least what I'm told]). But not to worry, we sorted that all out after I contacted them.

Also, threatening a journalist is probably not the best thing to do. This is probably one of the major reasons I decided to follow up and spend four days worth of research into this matter (Would have only been two, but an unfortunate power surge lost some research.)

In response to me telling him that I was unable to publish the info, he sent me this:

> Verify as you wish, but we expect something produced now for the information provided and our time spent.

I just want to note, just because you answered my emails doesn't mean I'm obligated to publish something. However, since you expect something to be published, this is what you're going to get.

Do the Math

In one correspondence, I requested wholesale prices. And here they are

70.00 per unit
100 pcs $5,500.00
250 pcs $13,700.00
500 pcs $25,000.00
1000 pcs $46,000.00
5000 pcs $245,000.00
10000 pcs $500,000.00

At first glance, everything looks normal, but let’s do the math anyways.

5,500.00/100 = 55 USD per piece
13,700.00/250 = 54.8 USD per piece
25,000.00/500 = 50 USD per piece
46,000.00/1,000 = 46 USD per piece
245,000.00/5,000 = 49 USD per piece
500,000.00/10,000 = 50 USD per piece

I know it’s hedging on nit picking, but this helps demonstrate the quality of this outfit. Someone really needs to brush up on their math skills. By far, if I order 1,000 pieces I save 3 dollars from ordering 5,000 pieces. Normally, in business, a straight percentage is applied and often increases at a set interval. For example, 10% discount at 1,000 but 15% discount at 10,000.

Problems with the hostname
The next problem deals with the fact that even though the companies claim to be separate, an apparent link between them can be determined using the IP address contained in email correspondence. For obvious reasons, we will not disclose the IP address or hostname. We would simply like to point out that all emails coming from the multiple domains, originate from the same hostname. This hostname contains roadrunner.nf.net. Furthermore, an unnamed source from xbox-scene confirms that the hostname is the same as SkyD, who apparently uses IRC a lot.

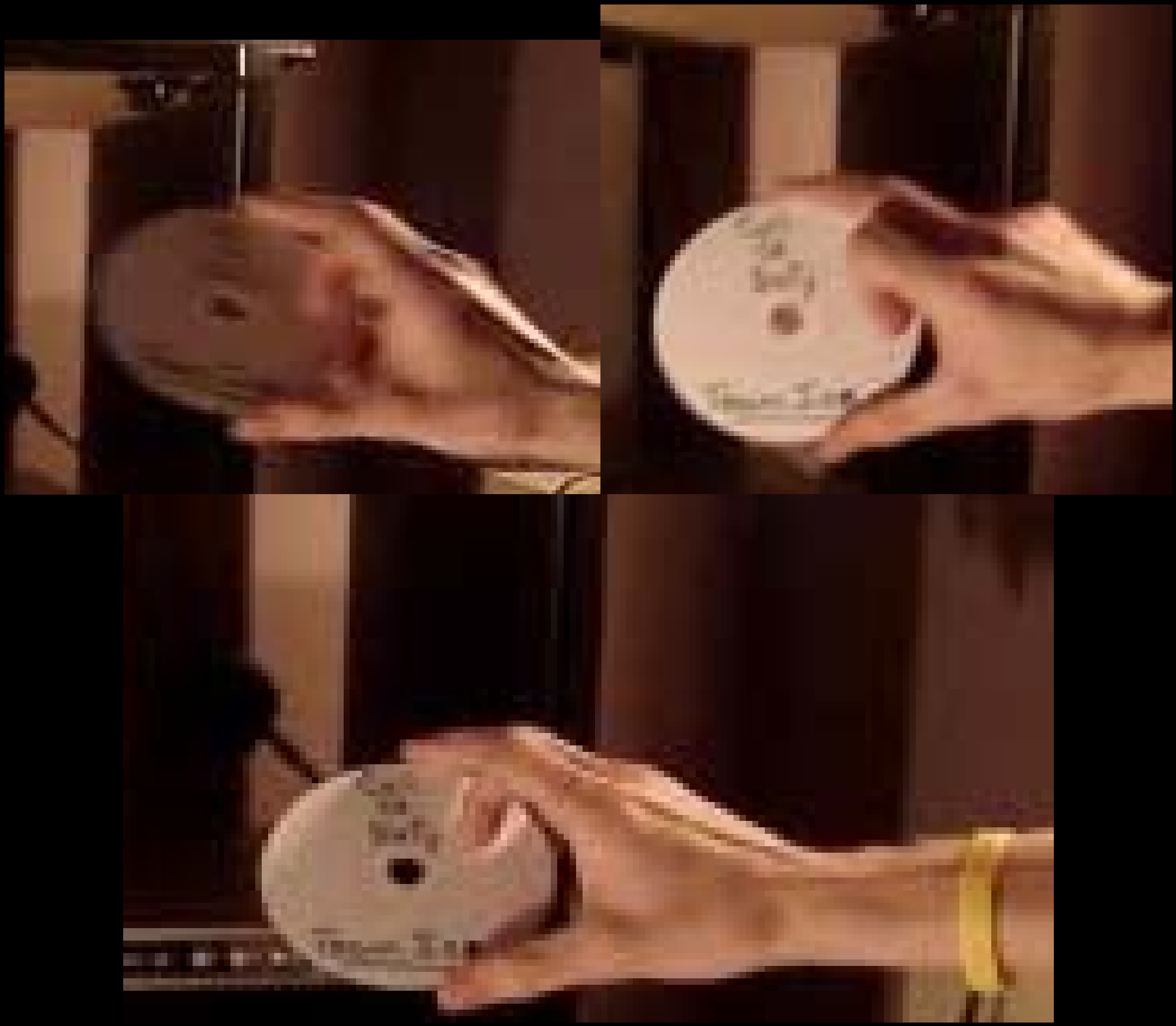
> “Usually any of the console channels on EfNet...he goes by the nick "Backups"”
>
> “I'm pretty sure he's in that area when I've seen him on IRC it's always the same hostname”

This was in response to a question about the actual location of the owner of icemodchip.com, infinitymods.com and rarecodes.com. The only thing I will divulge about the informant is that he has close ties with Xbox-Scene and is trusted by them.

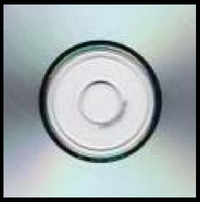
Addressing the New icemodchip.com video
Recently, icemodchip.com has released two videos. The first video was released and proved to be fraudulent after http://naoneo.com/ released a video showing that frames had been removed (Meaning that the video was edited and the person could have easily swapped the disks out.) The second video release becomes a little trickier to disprove, but with the right tools, we are able to provide enough evidence to cause doubt.

Our initial impression on the second video where mixed. On a positive aspect, no frames had been removed from the video. The only problem, which sent of red flags, was the way he handled the disc. In the previous video, he openly showed us the back of the disc to confirm it was burnt (As stated before, the video editing causes us to believe he paused the tape, swapped out the disc and the started recording again [In the first video]). The way he handled the disc in this video is somewhat obvious to the trained eye. At no point did he allow the back of the DVD to be caught on camera. In fact, it appears that he purposely changed hands in order to prevent the readable surface from being shown.

Upon further investigation, we quickly came to the conclusion that the DVD was simply a real Call of Duty 2 disk covered by a cheap CD printing label. The reason for our conclusion came from maybe a dozen of frames where the light hitting the DVD caused the label to become darker than the inner circle. This WILL NOT happen if the media was silk-screened or any other means of commercial printing. The only plausible explanation is CD labels. These labels allow light to pass through depending on whether the surface it is attached to is transparent. The following frames have been included to show the CD label is indeed covering a valid COD2 disc.



At first, you may not notice, but carefully inspect the inner circle. On a typical blank DVD or CD, this area would be clear. If the disc in question is covered by a printable label, the transparent or holographic part would cause the area to become lighter when light hits it a certain way.



Also, arguments have been made that DVD and CDs used for printing could have easily been used. This is indeed false. The technique used to apply the base surface to the CD would not cause the inner circle to become lighter when a darker background is applied or when light hits it a certain way. Furthermore, no commercial or consumer "printable" DVD or CD would cover the entire disc to the central hole. Also, notice the dark circular ring that appears around the inner circle. This evidence leads us to believe that the disc being used is nothing more that a Call of Duty 2 disk covered by an inkjet CD/DVD printing label. Also, naoneo agrees with us by saying

> *"Well, realistic if you are a retard, there are no DVD-R media on this planet that cover the disk right to the central hole, now, why does theres?, could it be that a retail disk hides beneath that white paper?, why cover the central right you ask?, could it also be thats because retail disks have clear marking on the hub of the disk, that would have, undoubtably given away their stupid plot? ANYONE, can stick white paper on a retail disk, write some crap on it and make a video pretending to be booting a burned disk, Anyone."*

Just to note, enhanced video showing the color differences may be released later on.

Can we meet?

Wanting to give icemodchip.com every chance at redeeming themselves, I gladly offered to fly up and view the modchip demonstration. The only conditions that I insisted on was that I would have to examine the "Copied" game and have the Xbox 360 open and all connections to the display in clear view. I told him that no photos would be taken, and no attempt to copy or duplicate his work would be made. Furthermore, I would not disclose his location or personal details. Upon this request, he sent us the link to the video. I complained and told him that the video did not satisfy my doubts and that I would be willing to fly up on January 13th and 14th, view the modchip in a working environment, and fly back on the 15th to release my research. I also stated that I would not publish my work since I still doubted the actual existence of such work. On this claim, he sent me this

> No thanks wall street journal is interested.

If he isn't willing to show me his work and allow me to verify it, why would the Wall Street Journal publish anything?? Furthermore, I think the Wall Street Journal would require some sort of proof. Our friends over at gamesfirst.com (An excellent source for gaming news) had also contacted icemodchip.com and offered to send a reporter to verify. In a recent article, they provide this information for us

> January 7th, 2006: GamesFirst.com sends Team I.C.E. an e-mail offering to send a writer to help verify that the chip was able to do what Team I.C.E. claimed. Team I.C.E. replied that they, "Hey we would love to but we are currently covering press with divineo at the moment over in france :) Stay tuned to the website, we will be posting os pictures soon!"
>
> January 10th, 2006: Team I.C.E. continues to be defiant, saying that they will be vindicated when the chip launches. Subsequent requests to allow a writer to verify the chip are ignored, or not addressed in replied e-mails.
>
> http://gamesfirst.com/index.php?id=1141

I would highly suggest reading their report and research since they cover a few topics that we chose not to cover.

Divineo rumors

Recently, divineo provided this information to the public:

> "The 360 chip is 99% a scam, we had booked tickets to go meet them, we were ready to make huge prepayment and more after we meet the guys at their office and see the working X-Box 360 with back-ups they 100% confirmed they had and it was no problem to show etc. Now at last second they cancelled, with some lame reason, those guys are scammers I really believe. They did the same to Xenium, backed off and changed completely what they said." – From http://ps2nfo.com/

Just in case you may be new to the scene, Divineo is one of the top respected modchip dealers around. We would like to point out one apparent problem with the info provided by Divineo and icemodchip.com.

Divineo

> "…we were ready to make huge prepayment and more after we meet the guys at their office and see the working X-Box 360 with back-ups they 100% confirmed they had and it was no problem to show etc."

Icemodchip.com

> "we are currently covering press with divineo at the moment over in france"
> Provided by gamesfirst.com – Jan 7th

Divineo says that they intended to meet Team ICE at their office, which is in Canada. Why would they require Team ICE to fly to France?? Unless icemodchip.com is in France (And we are 99% sure they aren't), there would be no reason to meet there. In response to Divineo's bad press, icemodchip.com responded with

"Divineo wished to buy us out, at which time, we refused, so they choice to try and slander our name." – gamesfirst.com

Now I doubt Divineo would offer to purchase their company without first verifying their claims. Also I tend to believe Divineo since they have a long-standing reputation with the console modding community.

Conclusion:

Well I promised to release this before the end of January the 16th, and in order to do so, I removed SOME information that was repetitive or unnecessary. Mainly the fake modchip pictures and a few others are not talked about. I highly suggest you read gamesfirst.com and naoneo.com for more information (http://67.15.28.13/~naoneo/naoneo/whyfake.php). Blacklisted411 has no official position on the modchip and will continue to do so until icemodchip.com either provides verifiable proof or allows us to view a working product. We hope that this information will provide you with insight into the claims made by icemodchip.com. If icemodchip.com releases any new information or videos, a short analysis will be released by Blacklisted411.net in order to combat any concerns that may arise while viewing it.

NOTICE: If you would like a copy of the email to verify the contents of our quotes, we will provide one on request.

License: Blacklisted 411 grants anyone the right to republish the whole document and or parts of it as long as the following conditions are met:

1. Proper credit is given to Blacklisted411.net and the author

2. No money is to be charged for the document (This does not include subscription news services and magazines as long as the subscription or magazine is not solely based on the contents of the article.)

The information contained in this article to many hours of research and we would appreciate it if proper credit was given. If you have any questions, please feel free to contact the author. Extended research may be released in the future in order to satisfy public demand, but we feel that the above information will suffice for the meantime.

Ustler

Ustler@blacklisted411.net

[c]

Communication: Coded Conversation

By Israel Torres <israel@israeltorres.org>

<Action!>
Frank: *"Hey Fred did you see Sally at the party last night?"*
Fred: *"Are you kidding me? You'd have to be dead to not notice her…"*
<Sally walks in>
Sally: "Notice her what? Who's what?"
<Long awkward silence>
Frank: "Oh hey Sally, we were talking about the game on TV."
Fred: "Yeah the game on TV."
Sally: "Game? Hey! Were you guys talking about me? Was it about last night?"
<Fade to black>

I am sure that you have been in a similar scenario more than once before. **Let's break it down.** We have at least two individuals that wish to communicate privately in a public space. Given that they are speaking a common language not unfamiliar to the public space, they really can't say what they want to say too loudly without giving away what they are talking about to any passerby. In fact they may draw suspicion if they lower their voices or altogether stop their communication abruptly. There are times and places where such behavior could cost your life, or even your freedom.

Those that are bilingual or multilingual may be able to afford dropping into another language or dialect when presented with unwanted company as is also normally practiced. However this also brings attention and the listening parties usually get the idea that they are being talked about (or are attempting to hide something) – from their perception why else would they suddenly switch into another language?

Those that aren't bilingual or have an alternate common language between one another usually just panic and stop talking when the other party comes within listening distance. Again this brings attention that something is up. It may not be anything out of the norm, but in the back of the party approaching they become extra aware because of this. Perhaps in the next time they will be more stealth in their approach.

Being put into a situation where any type of communication becomes awkward is only proof that you have not prepared for **coded conversation**. Coded conversation is communication between given parties that has been modified to disallow unwanted listening parties from understanding what is actually being communicated and yet at the same time not **letting on** that the communication itself is being modified for this purpose. This is different than encryption because the usage of encryption is often more obvious that it is being used than not (*think static and garble*). Coded conversation is closer to the art of steganography where within the obvious message there is a "secret" hidden message. One main difference between either of these studies is that coded conversation is done by planning ahead to create a protocol that will be done without the usage of computer calculation or such technologies to apply this protocol. **Yes we are talking about old school brain power – batteries not included.**

Here we go back to the history lesson (*I can hear you groaning*). A critical time where communication really is important is at time of war. If a message doesn't get to where it is supposed to in time **bad** things will happen. Also if a message is intercepted by the enemy and something learned from this interception **worse** things could happen. It is a constant battle to figure out the adversary's codes while at the same time protecting your codes at all cost from the adversary. It is also understood that the first "side" to crack the others code earns the advantage for victory. Such practices were in their infancy at the time of the World Wars. Codes were being broken left and right by some of the brightest minds on all sides. You have to figure that if one human brain can do something there should be at least another one out there that can undo it – it is human nature.

Back in the day the way a message was coded was kept at an unfathomable secret level because if the adversary knew how the message was built, they could reverse the process easier other than attacking it through brute force. It is very different with today's technology but still a simple method prevails even in the age of the distributed iron. This is known as One Time Pad (OTP). This simple method requires the sender and the recipient to understand the protocol the message is being sent in as well as establishing a meaning to what is being sent – regardless of what is being sent. In the meanwhile the enemies will be confounded with trying to find meaning in what is being sent so they can eventually crack the code. The downfall is when either the sender or recipient lose or reveal their codebook the mechanism breaks. To make up for this the sender and recipient must be prepared and trained with different instances of these codes. The real trick is never to record them in any other way than memory commitment. With OTP if the enemy can't find a code book, it just can't be broken without a mind-reading machine. The best part about this (code talking) is that it is VERY low technology – no computers required.

The downfall with code talking in public is very similar to that of speaking another language in public – it calls attention to you because you are now obviously speaking something uncommon to the commoners. Though we can extrapolate the ideas used in **code talking** and apply them to **coded conversation**. With thinking ahead you and your party may create a protocol, code words and code phrases that will be invisible to anyone listening to your conversation. It can be useful whether you are being listened to by electronics surveillance or the naked ear.

Just to make it clear we aren't talking about coding such as 10-codes used by CBers and law enforcement units to communicate with abbreviated substitutions such as 10-4 being recognized as OK or acknowledged. We also aren't talking about using phonetic structure such as "Tango Down", obvious coding such as "The egg is in the nest". Other forms also not used in this circumstance are forms of dialect, slang or vernacular. **Coded conversation is a stealth substitution of meaning other than what is being heard by unwanted parties.** There is no way to “crack it” in real-time where it would yield more value than at any other time. If done well enough all meaning could be lost with time thus making any other method useless in attack. The overall idea would be that **there is no code.**

Such a protocol would need to address the following methods common to parties to be involved:

- Code enablement (*giving notice that the coded conversation is beginning and will be persistent until further notice* [code disablement])
- Coded messages (*the payload message or messages [word/phrase substitution]*)
- Code disablement (*giving notice that the coded conversation has ended and all conversation if any will not be coded*)

To keep stealth all coded conversation should not immediately deviate from the conversation at hand. Otherwise this will bring a notable period in time where an edge was created which could be noted at the time of enablement by undesirable parties. Message stuttering (confusion during the coded message) should also be avoided as it could bring notice that the conversation is being coded. Message stuttering is recognized when one of the parties lapses the protocol or does not recall the message response and errs with a question directly implying the code itself. (e.g. “what does that mean?”)

Let’s bring Frank and Fred back into another scenario using coded conversation:
<Action!>
Frank: “So **Fred** how about the game last night?” (**Code Enabled**)
Fred: “Oh it was something else. I lost 60 bucks!” **(Coded Message)**
Frank: “Are you serious? Haha you suck. There is no way I would have put 60 bucks on that team. Have you told Sally?” **(Coded Message)**
<Sally walking in>
Sally: “Told Sally what?” (**Interceptor – Bad Guy**)
Fred: “**Frank** and I were planning a surprise for you – so you are going to have to wait it out” (**Code Disabled**)
Frank: “Yeah, Sally, don’t ruin it.”
Sally: “OK you guys… you know I love surprises!”
<Fade to black>

Notice that coded conversation should have enough forward design to allot for unknowns. Handling unknowns is very important as they generally come up when least expected – expect them!

In this example of coded conversation Frank and Fred built their activation/deactivation protocol by saying the name of the person they are talking to. Once someone has said their name in their coded party the code is enabled. Anyone in the coded party can now disable the coded conversation by saying the name of anyone else in the party. Frank and Fred also had created substitutions for these words and phrases in the following manner:

Word/Phrase	**Substitution**	**Meaning**
Game	Stripper	Fred picked up a local stripper while Sally sat on AOL all night again.
Something Else	Unwanted	The stripper up and left after I handed her the money. I didn’t get to hit it.
60	600	A tens multiplier – Any number said is automatically multiplied by 10.

Surprise	Boys night out	More time away from Sally.

Anyone listening would have just assumed or even analyzed that Fred and Frank are talking about last night’s game (Yeah they are smart enough to know if a real game actually went on) and how Fred put 60 dollars on his team and lost it. In reality to Frank and Fred this isn’t the case at all. They have successfully used coded conversation to hide their meaning without letting on to anything that they are hiding anything.

Coded conversation also works great in situations where eye contact or body language cannot play a part in communication – such as over a cell phone. The key is to plan ahead and not let anyone outside of the coded party understand the meaning of the protocol or substituted word/phrases used.

Another great example of coded conversation is when one of the parties is put under duress and is coerced into trying to bait the other party so they may be revealed. The captive party could set up their coded conversation protocol to announce they are alone when in reality after they have activated the protocol the coded word “***alone***” means ***“Warning: I am under duress, expect company activate plan B”.*** To the hostile it would seem completely normal that the party they are holding captive is stating they are alone and not be aware that the remote party has now activated their defenses.

Coded conversation takes preparation, memorization and forward thinking to use successfully. Remember never write anything down or record messages otherwise where they may be found and later used to break the code – *or used against you*. It is also wise to enable alternate methods of speaking a meaning so that conversations recorded by a third party, can resist cracking over time due to repeated usage.

When no one other than the intended recipient is aware of the coded conversation is when you have mastered coded conversation. Remember that next time you walk into a conversation in a public place.

Keeping it ‘rael,
Israel Torres

References and Related Links

“- to conceal the actual meaning of a conversation
- to conceal the fact that there is a hidden meaning “
http://www.speechcode.com/speech-code.html

*“A **speech code** is any rule or regulation that limits, restricts, or bans speech beyond the strict legal limitations upon freedom of speech or press found in the legal definitions of harassment, slander, libel, and fighting words.”*
http://en.wikipedia.org/wiki/Speech_code

“Newspeak is a fictional language in George Orwell's novel Nineteen Eighty-Four.”
http://en.wikipedia.org/wiki/Newspeak

"The Eagle Has Landed"
http://en.wikipedia.org/wiki/Neil_Armstrong

"The egg is in the nest."
Mars Attacks
http://project.insysltd.ru:8080/pls/ivan/Show_Image?ImageId=33192

One time pad
http://en.wikipedia.org/wiki/One-time_pad

Navajo Code Talkers
http://www.history.navy.mil/faqs/faq61-2.htm

Navajo Code Talker Dictionary
http://www.history.navy.mil/faqs/faq61-4.htm

Windtalkers (The Movie [2002])
http://www.imdb.com/title/tt0245562/

Code Talker
http://en.wikipedia.org/wiki/Code_talker

NATO phonetic Alphabet
http://en.wikipedia.org/wiki/NATO_phonetic_alphabet

Ten-codes
http://en.wikipedia.org/wiki/10-codes

Q-code
http://en.wikipedia.org/wiki/Q_code

***"Steganography** is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message"*
http://en.wikipedia.org/wiki/Steganography

"**Spam Mimic** gives you access to a program that will encrypt a short message into a message that appears to be spam."
http://www.spammimic.com/

Sams Big Play Maker is a Win32 program that converts arbitrary text to what appears to be a play script.
http://www.scramdisk.clara.net/play/playmaker.html

[4] ==Cryptogram==

January's Cryptograms

By Israel Torres <israel@israeltorres.org>

You have two cryptograms to solve. The first one is considered to be for beginners (A), and the second one (B)is for intermediate to advanced aspiring cryptographers.

Cryptogram A (*beginner*)
5EBE2294
ECD0E0F0
8EAB7690
D2A6EE69

Cryptogram B (*advanced*)
1814E0A5
AF374437
03D7EE4C
87EFA6BA

Hint: There is **no** need to GET the SID to solve this cryptogram. You also shouldn't need Bravo Foxtrot.

Your mission if you choose to accept it is solving both cryptograms (if you can). To qualify as a winner you need to submit two things:

1. The solved message
2. How you solved the cryptogram by describing each step, algorithm name, etc. The more information you submit the more we'll believe you.

Send both items 1 and 2 to crypto@blacklisted411.net with the subject of "cryptograms"

Winners will be posted in order of submittals based on correctness and completeness above. If you can only solve one cryptogram that is OK as well but doesn't have as much weight as someone who solves both of them.

Winners will be posted right before the next edition is released.

Keeping it 'rael,
Israel Torres

[5] ==Favorite Photo==

Im sorry, but this is just wrong. Federal police force? Hmm..
http://infowars.com/articles/ps/patriot_act_empowers_homeland_security_gestapo.htm

[6] ==Credits==

Chief Editor: Zachary Blackstone

Online Edition Editor: Ustler

Authors: Ustler, Israel Torres