



Easy2.Net 유흥관

 decrypt_message	2013-10-27 오후 1...	텍스트 문서	1KB
 POWEROFXX_easy_reversing2	2013-10-27 오후 1...	응용 프로그램	305KB

압축을 풀니 두 개의 파일이 있다. decrypt_message 파일을 보니 입력한 string을 암호화해주는 프로그램인 것 같다.



한 글자 씩 숫자로 나타내어 주는데, 입력하는 것에 따라서 다르게 나온다..

```
Sweet Secret Memo _ logger 0.1
Date : 10.26_23:44 [UTC + 09:00]
Chicken's Message : 144 207 93 170 111 25 130 144 21 152 172 -3 113 164 -24 94 117 -25 173 106 77 169 119 15 135 127
9 111 123 -12 109 218 61 155 98 50 199 131 -2 168 205 75 169 164 24 139 116 95 127 113 71 102 141 -18 152 153 28 111
131 74 207 201 -21 174 155 74 138 164 -7 179 215 94 187 142 55 147 99 64 176 117 25 193 177 47 135 111 -5 108 137 -8
132 120
Egg's Message : 210 209 22 126 188 29 212 101 24 125 145 -22 180 134 -30 162 107 18 142 210 8 185 95 -23 221 123 54
183 180 39 103 145 57 141 181 56 141 138 -25 124 133
End of Log
```

decrypt_message에 있는 이 두 개의 암호를 복호화하면 flag가 나올 것 같다.

ilspy로 파일을 디컴파일 해본 후 분석을 해보았다.

```
public string d8jergu394r0nnsjd94jfs = "9pMaVs5Dxi0PGe8JETXyMg3lbudro6Qk1WLKwyhfnS4Iv0ABtjUCc7RZz2NFHq";  
public int njgdcxdxxx6r = DateTime.Today.DayOfYear;  
public int zfgvjnkji8y6ug9u9i = DateTime.Now.Hour;  
public int cljbyt798ygdre5 = DateTime.Now.Minute;  
public int zsawrsrf6g0i98t6vllp = DateTime.Today.Month;  
public int qexyg8j9u8thuhg = DateTime.Today.Day;
```

변수명 정말 슬퍼요..

d8jer에는 문자열이 무작위로 들어있고, 밑에 부분을 보니 현재 시각을 받아오는 것 같다.

```
public Form1()  
{  
    this.InitializeComponent();  
    this.d8jergu394r0nnsjd94jfs += "KfeR0dEILJs5W6D1m4XFtH7YbwgrUConPuqQBcSxT092z1jv8yMAGhpZN3akVi";  
    this.d8jergu394r0nnsjd94jfs += "8vxeKVPpYlsXDAujWoJEingTGf3mCh59LR0t6cdUNMb41zH7Kr0yS2BIFZqawQ";  
}  
private void Form1_Load(object sender, EventArgs e)  
{  
    this.panel2.BackColor = Color.Transparent;  
    this.d8jergu394r0nnsjd94jfs += "Hv8VzYa5b1FMGNODW4kwX9L3hK6SqsTtyxoE0Z7fPJIGrCAQiljBuenRcp2dUm";  
    this.d8jergu394r0nnsjd94jfs += "fgnCw4HPJRdXKIq31YNDZMS820jA7eUxpozASvmykiQrTFLW6htGb9B01EcVu5";  
    this.d8jergu394r0nnsjd94jfs += "jxLaZdwYngAfKGNhzTcXQU7Jy9sFbp0eRI1ECrv23PSw846oH5MBVt1DiOqumk";  
    this.d8jergu394r0nnsjd94jfs += "U0tnl9bVK4iB2LzZXy7PaChCAI5p0sSfjgqkr1vuRTFEo8Dxmhw3QGdeJM6wYN";  
    this.d8jergu394r0nnsjd94jfs += "gWYN9w4LuPjxJl1Mh0kniQy8CBUXr6THaKDctEdb0Imp32VffZGvAS5ezqsR7o";  
    this.d8jergu394r0nnsjd94jfs += "wj3J9fL8QY2kArXKgOEzmSdqHpcMsn1ahGwxCe7yPiLTuDRb6F40oZtiUBvV5N";  
}
```

밑에서 찾은 것들인데 d8jer은 정말 많은 고비를 겪는 것 같다. 다 사용할 진 모르겠다.

```
private void button1_Click(object sender, EventArgs e)
{
    if (this.textBox2.Text.Length > this.d8jergu394r0nnsjd94jfs.Length - this.zsawsrnf6g0i98t6vllp)
    {
        MessageBox.Show("Length Error", "ERROR_bb");
        return;
    }
    this.gettime();
    string text = this.h8r9gu4inheiprhgncvjousdfgiuweg(this.textBox2.Text);
    this.textBox1.Text = text;
    this.lfkfidngigiwhiu3yr89igorg(text);
}
```

버튼을 클릭하면 gettime 함수로 현재의 시간을 받아오고, h8r9gu~~ 함수를 실행하는데

```
public string h8r9gu4inheiprhgncvjousdfgiuweg(string str)
{
    StringBuilder stringBuilder = new StringBuilder();
    string text = this.ldfogndkfvisgi490rjgdijgw434ref(Form1.ncfjgirerg430t34trdgdfs(str));
    string[] array = text.Split(new char[]
    {
        ' ',
    });
    for (int i = 0; i < array.Length - 1; i++)
    {
        stringBuilder.Append(this.kfig9jepoingndkfvdjroger(array[i], i) + " ");
    }
    return stringBuilder.ToString();
}
```

h8r9gu 함수는 암호화를 해주는 함수이다. str을 받아서 보내고 저장하고 하는 것을 보면 여기에서 총 2번의 암호화가 실행된다.

```
public static string ncfjgirerg430t34trdgdfs(string input)
{
    int length = input.Length;
    char[] array = new char[length];
    for (int i = 0; i < input.Length; i++)
    {
        array[i] = input[length - i - 1];
    }
    return new string(array);
}
```

문자열을 ldfo에 보내기 전에 ncfj 함수로 문자열을 뒤집는다.

```

public string ldfogndkfvisgi490rjgdijgw434ref(string a)
{
    StringBuilder stringBuilder = new StringBuilder();
    int num = this.zsawsr6g0i98t6vllp;
    char[] array = a.ToCharArray();
    for (int i = 0; i < a.Length; i++)
    {
        stringBuilder.Append((int)(this.d8jengu394r0nnsjd94jfs[i + num] ^ array[i]) + " ");
    }
    return stringBuilder.ToString();
}

```

ldfo 함수를 보면, num에 현재 월(month)를 저장하고, 입력 받은 문자열을 array 배열에 차곡차곡 배열화 시킨다. 그리고 a의 길이만큼 d8jer[i+num]의 배열과 ^ 연산을 한 후 공백을 추가시킨다.

여기서 나온 결과 값은 다시 h8 함수로 돌아가서 공백을 제거 된 후 배열에 들어간 다음 kfig 함수의 암호화를 실행하게 된다.

```

public int kfig9jepoingndkfvdjroger(string chr, int range)
{
    int num = int.Parse(chr);
    int num2 = range % 3;
    int num3 = 2;
    if (num2 == 0)
    {
        num += this.qexyg8j9u8thuhg * num3 + this.cljbyt798ygdre5 * num3 - this.zfgvjnkji8y6ug9u9i * 2;
    }
    else if (num2 == 1)
    {
        num += this.zsawsr6g0i98t6vllp * 3 + this.cljbyt798ygdre5 * 2 - this.zfgvjnkji8y6ug9u9i * num2;
    }
    else if (num2 == 2)
    {
        num += this.njgcgcxdxx6r - this.zsawsr6g0i98t6vllp * (num2 * 5) - this.cljbyt798ygdre5 * num2 - this.zfgvjnkji8y6ug9u9i * (num3 + 4) - num2;
    }
    return num;
}

```

배열의 인덱스를 3으로 나눈 값에 따라서 암호화를 다르게 한다. 이렇게 해서 암호가 각자 들쭉날쭉 이었던 것 같다.

머리로 계산하는 것은 딱히 좋은 일은 아닌 것 같아서, 이번 문제는 저번 문제와 다르게 복호화하는 프로그램을 코딩해야할 것 같다.

A,B,C 순으로 암호화 했으면 복호화는 C,B,A 순서이므로 kfig 함수부터 복호화하는 코드를 짤 것이다. num += 이므로 num -= 로만 해주면 복호화가 될 것이다.

그런 후 xor 연산을 다시 해준 후 거꾸로 뒤집으면 될 것이다.

```
# Easy2.net decode.py

month = 10
day = 26
hour = 23
minute = 44
year = 299

chick = "144 207 93 170 111 25 130 144 21 152 172 "
chick += "-3 113 164 -24 94 117 -25 173 106 77 169 "
chick += "119 15 135 127 9 111 123 -12 109 218 61 "
chick += "155 98 50 199 131 -2 168 205 75 169 164 "
chick += "24 139 116 95 127 113 71 102 141 -18 "
chick += "152 153 28 111 131 74 207 201 -21 174 "
chick += "155 74 138 164 -7 179 215 94 187 142 "
chick += "55 147 99 64 176 117 25 193 177 47 135 111 -5 108 137 -8 132 120"

egg = "210 209 22 126 188 29 212 101 24 125 145 -22 180 134 -30 162 107 18 "
egg += "142 210 8 185 95 -23 221 123 54 183 180 39 103 145 57 141 181 56 141 138 -25 124 133"

decode_key = "9pMaVs5Dxi0PGe8JETXYmg3lbudro6Qk1WLKwyhfnS4Iv0ABtjUCc7RZz2NFHq"
decode_key += "KfeR0dEILJs5W6D1m4XFtH7YbwgrUConPuqQBcSxT092z1jv8yMAGhpZN3akVi"
decode_key += "8vxekVPpYlsXDAujWoJEingTGf3mCh59LR0t6cdUNMb41zH7Kr0yS2BIFZqawQ"
decode_key += "Hv8VzYa5b1FMGNODW4kwX9L3hK6SqsTtyxoE0Z7fPJIGrCAQiljBuenRcp2dUm"
decode_key += "fgnCw4HPJRdXKIq31YNDZMS820jA7eUxpozasVmykiQrTFLW6htGb9B0IEcvu5"
decode_key += "jxLaZdWYngAfKGNhzTcXQU7Jy9sFbp0eRI1ECrv23PSw846oH5MBVt1DiOqumk"
decode_key += "U0tnl9bVK4iB2LzZXy7PaCHcAI5p0sSfjgqkr1vuRTFEo8Dxmhw3QGdeJM6WYN"
decode_key += "gWYN9w4LuPjxJl1Mh0kniQy8CBUXr6THaKDctEdb0Imp32VffZGvAS5eqsR7o"
decode_key += "wj3J9fL8QY2kArXKgOEzmSdqHpcMsn1ahGWxCe7yPIlTuDRb6F40oZtiUBvV5N"
```

가장 먼저 필요한 변수들을 적어준다. year가 연도인줄 알고 만나와있어서 물어보려다가 Day Of Year를 보고 지금까지 지나온 날 인 것을 깨달았다.

```
def stage1(string, range):
    num = int(string)
    num2 = range % 3
    num3 = 2

    if num2 == 0:
        num -= (day*num3) + (minute*num3) - (hour*2)

    elif num2 == 1:
        num -= (month*3) + (minute*2) - (hour*num2)

    elif num2 == 2:
        num -= year - (month*(num2*5)) - (minute*num2) - (hour*(num3+4)) - (num2*num3)

    return num;
```

kfig 함수가 마지막에 실행되므로 거꾸로 거슬러 올라가면 될 것이다. +=를 해주므로 -=를 해주자

```
def stage2(string):
    num = month
    array = string.split(' ')
    result = ''
    count = 0

    for i in array:
        if i == "":
            break;
        array_num = int(i)
        result += chr(array_num ^ ord(decode_key[count+num]))
        count += 1

    return result
```

그 다음 xor 연산을 하는 부분은 다시 연산해주면 원래 값이 나온다. ord를 생각 안하고 int로 삽질하다가 아스키코드여야함을 깨달았다.

```

print "[*] chick key decode"
chick_array = chick.split(' ')
chick_len = len(chick_array)
chick_stg1 = ''

for i in range(0, chick_len-1):
    chick_stg1 += str(stage1(chick_array[i], i)) + ' '

print "stage 1 : " + chick_stg1

chick_stg2 = stage2(chick_stg1)
print "result : " + chick_stg2[::-1]

print "\n[*] egg key decode"
egg_array = egg.split(' ')
egg_len = len(egg_array)
egg_stg1 = ''

for i in range(0, egg_len-1):
    egg_stg1 += str(stage1(egg_array[i], i)) + ' '
print "stage 1 : " + egg_stg1

egg_stg2 = stage2(egg_stg1)
print "result : " + egg_stg2[::-1]

```

마지막으로 암호들을 복호화한다음 [::-1]로 뒤집어주면!

```

[*] chick key decode
stage 1 : 50 112 124 76 16 56 36 49 52 58 77 28 19 69 7 0 22 6 79 11 108 75 24 46 41 32 40 17 28 19 15 123 92 61 3 81 10
5 36 29 74 110 106 75 69 55 45 21 126 33 18 102 8 46 13 58 58 59 17 36 105 113 106 10 80 60 105 44 69 24 85 120 125 93 4
7 86 53 4 95 82 22 56 99 82 78 41 16 26 14 42 23 38
result : uth_key = md5(flag); auth_key = auth_key.ToLower(); if(auth(auth_key) == true) { clear(); }

[*] egg key decode
stage 1 : 116 114 53 32 93 60 118 6 55 31 50 9 86 39 1 68 12 49 48 115 39 91 0 8 127 28 85 89 85 70 9 50 88 47 86 87 47
43 6 30
result : tring flag = "W3_10vE_Ch1cKen_FoR3veEr";

```

플래그가 뜬다. 처음에 코딩 다하고 이상한 !=^~ 이런 문자열이 났는데 함수 부분엔 아무 문제가 없어서 엄청 고민했다. 이것저것 바꿔보다가 decode_key를 다시 입력해주니까 되더라.. 왜 그렇게 된건진 잘 모르겠다