ClevCode

VULNERABILITY RESEARCH, EXPLOIT DEVELOPMENT, REVERSE-ENGINEERING

Cicada 3301

For those of you still unfamiliar with the Cicada 3301 puzzle, it has been called “the most elaborate and mysterious puzzle of the internet age” by Metro, and is listed as one of the “Top 5 eeriest, unsolved mysteries of the Internet” by The Washington Post. Here are links to some of the articles that were written about me and my work on the original Cicada 3301 challenge back in 2012:

Meet The Man Who Solved The Mysterious Cicada 3301 Puzzle
The Internet Mystery That Has the World Baffled

Update (2014-01-05):

The integer sequence found in the first QR code felt familiar, but I was not sure why. I must admit that I did not give it much thought either, after ruling out any hidden ASCII message. Today I received a message from someone calling himself RoboSimian, with the following suggestion:

> 011358131345589144333776109871597584418167 65

> Not sure if it means anything, but at first glance the integer sequence contains sub-sequences similar in nature to the Fibonacci Sequence:
>
> 0, 1, 1, 3, 5, 8, 1 3
> and
> 13, 45, 58
> but I'm not sure what to make of the rest:
> 914433377610,987159758441816765

Note that the fibonacci sequence begins with 0, 1, 1, 2, 3, 5, 8 though, so the 2 is missing. Looking at the rest of the sequence, it turns out that it is in fact a perfect match with the fibonacci sequence, with all the 2:s removed! :)

So, the question is whether the sequence itself, the missing 2:s, or the continuation of the sequence, is the important part here. Note that there was a NUL-byte at the end of the decoded message, after the sequence. This could have some significance as well, suggesting that the sequence has been truncated, for instance.

Update (2014-01-05):

Two days ago I received the following message, from someone claiming to represent Cicada 3301:

> “When is a BWV not a BWV?
> You have gone farther than anyone.
> You have also been intelligent enough to identify false 3301s.
> Now I shall give you some answers. Our group is dedicated to re-cloaking privacy. You missed some hints. The Bach Mp3 was something that most hunters missed. 1033 BWV is not a Bach piece. It was composed by a man named Christoph. CPE misidentified the piece as his fathers..Pity. CPE was a teen, so forgive him. Seeing that so many false 3301's now join the game, we will post our next series of clues on a spoof website. We will be watching your work, since it is superior to anyone's work out there. Please treat this message with

> the same level of skeptical vision as any other errant email that you receive. Our goal, Clevcode, is to re-cloak humanity.
> Good luck,
>
> 3301

Someone also just posted this as a comment to this page:

> Secondly, has anyone bothered to vet, and I mean, really vet the Bach clue?
> I have.
> It's a ruse, and leads directly to 1033 BWV
> Further study leads me to believe Bach never wrote that piece.
> 3301 is leading us to a place where we doubt authenticity itself. We question experts.
> We patiently wait for a periodical cicada.

When digging into this a bit, many sources of information are found.
This is an excerpt from one of them:

> The Sonata in C major for flute and continuo, BWV1033, is preserved in a manuscript in the hand of C P E Bach, dating from the early 1730s, and in which he attributes the piece to his father. Its origins are obscure and disparate, perhaps since its first two movements, at least, are arguably more convincing as pieces for an unaccompanied melody instrument. Yet, in spite of sequential and cadential crudities, the music is not without either merit or charm and is, by and large, satisfying to play. There is a shapely nobility to the opening 'Andante', and a far from displeasing virtuosity, however simply conceived, in the ensuing 'Allegro'. The music of greatest substance, though, is to be found in the 'Adagio' which, like the concluding Minuets, 'alternativement', is not devoid of Bachian character. Bach's hand can surely be sensed, too, in the fully written-out parts of the first Minuet which bears relationship to a movement of a concerto by Bach's Merseburg contemporary, Christoph Förster; but, be that as it may, the sonata is uneven in quality and inconsistent in technique. It has been suggested that the harpsichord accompaniment was added later, perhaps by one of Bach's pupils.

The new message from Cicada 3301, or someone claiming to represent them, is probably suggesting that BWV1033 was composed by Christoph Förster rather than Johann Sebastian Bach.

Regarding posting their next series of clues on a “spoof web site”, this is probably it: Cicada 3301 releases clues to BBC News

When I first visited the link in question, I actually thought it was a real article. ;) Note that the real BBC site uses news.bbc.co.uk and not bbc-news-co.uk though.

Regardless of whether it is the real Cicada 3301 that has been sending these messages and releasing clues or not, considering that it is now January 5, I think it is quite likely that we will see a new Cicada puzzle being released very soon.. :)

Regarding the fake BBC site, I analyzed the “Within every image there is always a story” image, and found two QR codes.

The first one decoded to this string:

```
687474703A2F2F656D62656464656473772E6E65742F4F70656E507566665F53746567616E6F6772617068795F4
```

Decoded as a hex string, it yields the following message:

> http://embeddedsw.net/OpenPuff_Steganography_Home.html
> Sometimes the stories are hidden well, with many keys and many locks
> 01135813134558914433377610987159758441816765

The second one decoded to this string:

```
123157155145164151155145163040164150145040163164157162151145163040141162145040145166145156O
```

By looking at the patterns in this string, it is quite obvious that it consists of groups of three digits, which turns out to be octal-encoded. When decoding this string, we get:

> Sometimes the stories are even hidden to the maximum degree

For those of you who are curious how I performed the decoding, revealing the QR images was done using GIMP, by adjusting the color curves. That gave me the following image:

I manually stitched together the QR code in the corners to qr1.png, and inverted the colors, and decoded it using:

```
je@tiny:~/3301/qr1$ zbarimg -q --raw qr1.png | perl -pne 's{(..)}{chr(hex($1))}sgex'
http://embeddedsw.net/OpenPuff_Steganography_Home.html
Sometimes the stories are hidden well, with many keys and many locks
01135813134558914433377610987159758441816765
```

I placed the QR code from the top and bottom half of the picture in qr2.png, inverted the colors and decoded it using:

```
je@tiny:~/3301/qr2$ zbarimg -q --raw qr2.png | perl -pne 's{(...)}{chr(oct($1))}sgex'
Sometimes the stories are even hidden to the maximum degree
```

Since there is still no PGP signed message, it is doubtful that this is the real Cicada, but time will tell...

PS. The next logical step, if someone wants to analyze this further, is probably to use the OpenPuff steganography software, and maybe the "01135813134558914433377610987159758441816765" string as a key (hex-encoded binary data?), in order to extract yet another hidden message from the image on the fake BBC site, or from something else... Note that OpenPuff can use multiple keys, and multiple carriers (files) with hidden data.

I also received this message yesterday:

> Subject: Is this a message from Cicada?
>
> Hi there,
>
> I am a freelance content creator and recently came across some information regarding Cicada 3301, specifically the puzzle due to be released tomorrow, I have written an article on it and saw that you have already had some content on your site regarding it, I was wonder if you would be interested in putting it up on your site. I'm not asking for a fee in return but I would be grateful for a link through to my fiverr page.
>
> Please let me know if you might be interested.
>
> Thank you,
>
> Maria Jacobsen Holmes

With the follow-up message below:

> It wasn't actually a message that I received it was a request to write an article – I do freelance content creation, with specific words and information in it. It was also requested that I requested that I send it to you with the subject line I used. Would you like copy of the article?
>
> Maria

When I replied that I would like to see the message, I got this reply:

> It's just a general information post but I've put in bold the key words I was asked to put in before sending the article to you :).
>
> Oh and happy New Year!

The actual message (including the keywords put in bold) was:

> 3301 is a mysterious organization that has captured the attention of web aficionados since
> January 5th 2012 when a cryptic and baffling clue was released. Since then, on the same
> date each year, another clue has been released. What appears to be a simple image with
> some seemingly random phrases actually gives way to being one of the most
> sophisticated and well thought through web puzzles in the history of the internet and has
> attracted the attention of some of the most talented netizens of the world – fuelling
> rumours that it is a recruitment campaign for anything from governmental intelligence
> agencies to anarchistic hacker organizations. However the reality behind the puzzle is no
> where near close to being revealed and remains shrouded in mystery and left to the
> speculation of online chat rooms.
>
> The clues themselves demand a very interesting skill set; they don't just contain
> cryptograms and riddles but advance far out of the traditional domain of online puzzles to
> include historical themes, music, literature, poetry and much, much more. Although some
> major organizations, including intelligence ones, have used methods that bear some
> resemblance to 3301, nothing has ever come close to the scale and complexity of Cicada
> 3301; and the fact that so little is still known about the mysterious organization today just
> adds fuel to the fire that has captured global attention.

> The purpose behind these puzzles aside, online forums and chat rooms have exploded with users keen to share information and efforts in order to uncover the enigma behind this
> increasingly mysterious organization. Although to a large extent these conversations contain so many rumors, theories and speculations that they end up increasingly complicating matters, the usefulness of this information should not be ignored. There are a
> few recurring themes amongst this information, these combined with the new clues seem
> to be hinting that the focus for those wishing to solve the puzzle should expand to include
> **a historical element**. One mysterious figure has been brought to light and one wonders
> what the connection might be between **St. Germaine, an 18th century 'wonder man'** rumored to have been an incredibly powerful alchemist, a musical genius, friend of high
> ranking noblemen and kings, and perhaps most importantly immortal – and Cicada 3301.
> The recent clues have also been shifting the focus more towards music as a way to solve
> this riddle; the buzz on the chat room seems to indicate that musical leads hint towards a
> **Pythagoras connection**. Could this mean that 3301 are changing their delivery method and moving from TOR to actually **embedding their message into music**? Maybe the next set of clues released on January 5th will provide a little more clarity, but this in itself
> adds yet another element to the mysterious 3301 organization – **Why this date** and does
> the relevance of this date have something to do with solving the problem?

Note that the comment I received about the Bach clue was from someone calling himself "Germain"...

Update (2013-11-29):

I think this is probably an imposter, rather than the real Cicada, but I got a cryptic message after the article in Daily Telegraph, from someone calling himself Tibiceninae (the name of a cicada subfamily)... Unfortunately I don't have the time to look into it much deeper myself at the moment, but I have collected my notes on it so far here: http://www.clevcode.org/3301/

On January 4th 2012, an image was uploaded to various image boards, possibly originating at the infamous /b/ board at 4chan. When I came across it, I didn't think much of it at first, but still decided to look into it just in case it turned out to be interesting. I have always had a hard time resisting a challenge. This is the image that was posted:

My first thought was that it used steganography to hide a message, and since it was a JPEG image I tried using stegdetect by Niels Provos in case one of the detectable schemes was used. Since stegdetect have not been updated in almost 7 years, I didn't really get my hopes up that high though, but it is always worth a try. ;) The result can be seen below:

```
je@isis:~/3301/stage_1$ stegdetect 3301.jpg
3301.jpg : appended(61)&lt;[nonrandom][ASCII text][TIBERIVS CLAVDIV]&gt;
```

It did not detect any of the common steganographic schemes, but notified me of 61 appended bytes of ASCII text. Since my next move would have been to use “strings”, I would have discovered this anyway, but stegdetect was kind enough to tell me directly instead. :) So, let’s see what we have:

```
je@isis:~/3301/stage_1$ tail -61c 3301.jpg
TIBERIVS CLAVDIVS CAESAR says "lxxt>33m2mqkyv2gsq3q=w]O2ntk"
```

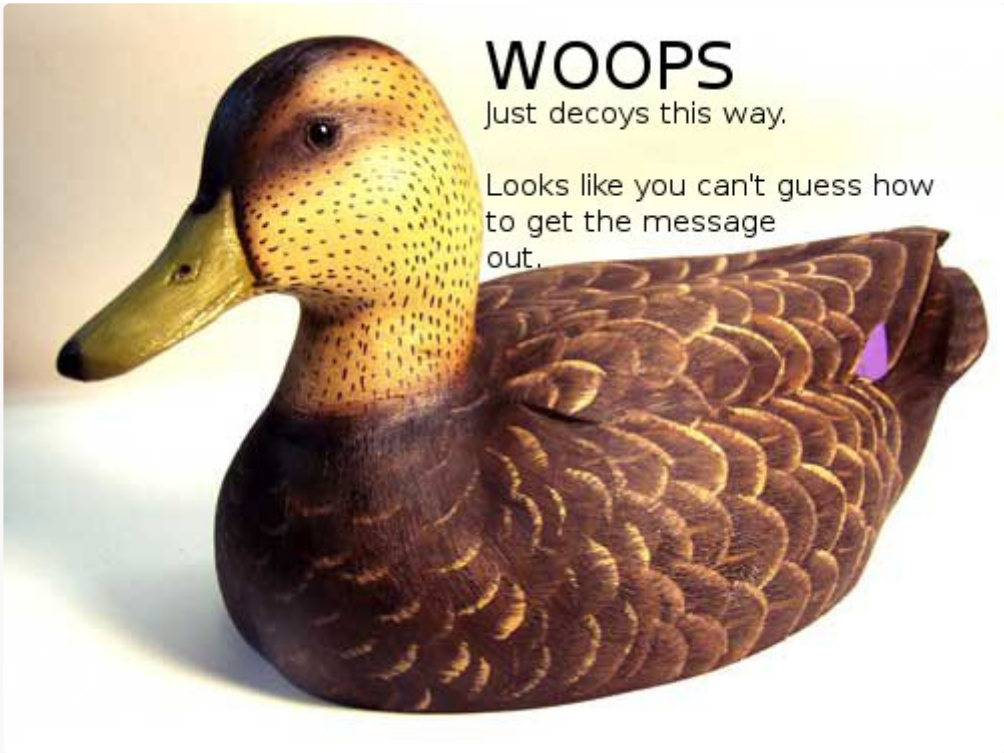
This is quite obviously a shift cipher of some sort (also known as a Caesar cipher), with “lxxt>33” being the ciphered version of “http://”. A shift cipher replaces each letter in the plaintext with a letter (or in this case, arbitrary ASCII character) with a letter a certain number of positions down the alphabet. So, let’s compare the ASCII values for the cipher text with the ASCII value of the supposed plaintext to see what the shift value is:

```
je@isis:~/3301/stage_1$ perl -e 'print ord("h")-ord("l"),chr(10)'
-4
```

In this particular case, this might have been a bit overkill, since we could just as well have manually counted the distance between h and l in the alphabet. ;) It is probably not a coincidence that Claudius happens to be the 4th Emperor of the Roman Empire, and the shift value happens to be 4, either. To decipher this, a perl oneliner is enough:

```
je@isis:~/3301/stage_1$ echo "lxxt&gt;33m2mqkyv2gsq3q=w]O2ntk" | perl -pne 'chomp;s{(.)}{ch
http://i.imgur.com/m9sYK.jpg
```

The image at the URL above can be seen below:



It seems like the challenge is a bit harder than a caesar cipher after all. Note that the message contains the words “out” and “guess” though, which could be a hint that we are actually supposed to use the old OutGuess tool to extract the hidden message. Incidentally, OutGuess is also developed by Niels Provos and is available for download from the same site as stegdetect (http://www.outguess.org/). Unfortunately, it seems like stegdetect is only able to detect when the older OutGuess 0.13b has been used and not OutGuess 0.2 (from 2001!). :D

Using outguess 0.2 with the -r option immediately reveals the hidden message in the original image:

```
je@isis:~/3301/stage_1$ outguess -r 3301.jpg 3301.txt
Reading 3301.jpg....
Extracting usable bits:   29049 bits
Steg retrieve: seed: 228, len: 535
```

The hidden message can be found here.

Now things are actually getting interesting. Although the challenge have not been required any particularly advanced skills yet, someone has obviously been putting some work into it. The hidden message says that we should go to the following URL:
http://www.reddit.com/r/a2e7j6ic78h0j/

The hidden message also includes a so called book code, consisting of a number of lines with two digits separated by a colon on each. The book and more information should be found at the URL above. Book ciphers are ciphers that use a book or a text of some sort as the key to encode a secret message. Traditionally, they worked by replacing words in the plaintext with the locations of words from a book, but in this case it seems more likely that the two digits separated by a colon in the code refers to a line and column number.

When visiting the Reddit page, we can make a number of observations. Most notably, there are a number of posts by the pseudonym CageThrottleUs that seem to consist of encoded text, which we can assume to be the book. It looks like an ordinary Caesar cipher may have been used, but on a closer look no shift value results in readable text. It seems most likely that a key of some sort is required to decode the text.

Looking closer on the page, we can see that the title is “a2e7j6ic78h0j7eiejd0120”. The URL itself is a truncated version of this. To the right, below the “subscribe” button, the title text is repeated and “Verify: 7A35090F” is written underneath. We can also see pictures of some mayan numbers on the top of the page. Mayan numbers are quite logical, at least from 0-19. A dot equals one, and a vertical line equals five. Two lines thus equals ten, one line with two dots equals seven (5 + 2) and so on. There is also a symbol resembling a rugby ball that equals zero. :)

The number sequence that is written using mayan numbers is as follows:
10 2 14 7 19 6 18 12 7 8 17 0 19

Comparing this with the a2e7j6ic78h0j7eiejd0120 in the title, we can see that numbers below 10 in the sequence above is also found in this string, at the same positions. Also note that instead of 10 we have “a”, instead of 14 we have “e”, and so on up to “j” being 19. Since the title of the page contains 23 characters and there were only 13 mayan numbers is is quite likely that we are supposed to continue converting characters from the title to numbers. This gives us:

10 2 14 7 19 6 18 12 7 8 17 0 19 7 14 18 14 19 13 0 1 2 0

This could very well be the key required to decode the text. Regarding the “Verify: 7A35090F”, it may refer to any number of things. A PGP key ID is, however, a good assumption since it consists of a 32 bit value normally encoded as eight hex characters and since PGP keys can be used to verify the signature, and thus the authenticity, of messages

signed with a PGP key. This could be quite handy, in case the challenge goes on and in case people decide to drop false leads to the people working on it. So, let's try to import the public key with the ID in question from one of the common PGP key servers:

```
je@isis:~$ gpg --recv-keys 7A35090F
gpg: requesting key 7A35090F from hkp server keys.gnupg.net
gpg: key 7A35090F: public key "Cicada 3301 (845145127)" imported
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
```

The comment for the key mentions 3301, which was used as the signature in the original image. It also includes the word "cicada" and the number 845145127, which may turn out to be significant at a later stage. Note, for instance, that cicadas emerge from their hideouts under earth every 13 or 17 years depending on which kind. By emerging every N:th year, where N happens to be a prime number, cicadas actually minimize the possibility of synchronizing with the life cycles of birds and other animals that prey on them. Also note that 3301 is a prime, and that 845145127 has 3301, 509 and 503 as its prime factors.

```
je@isis:~$ factor 3301
3301: 3301
je@isis:~$ factor 845145127
845145127: 503 509 3301
```

When taking a closer look at the lines of encoded text posted to the reddit page, we also find two images. One named Welcome and the other one Problems?. By using OutGuess again, we find another couple of hidden messages:

```
je@isis:~/3301/stage_2$ outguess -r welcome.jpg welcome.txt
Reading welcome.jpg....
Extracting usable bits:   326276 bits
Steg retrieve: seed: 58, len: 1089
je@isis:~/3301/stage_2$ cat welcome.txt
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

- From here on out, we will cryptographically sign all messages with this key.

It is available on the mit keyservers.  Key ID 7A35090F, as posted in a2e7j6ic78h0j.

Patience is a virtue.

Good luck.

3301
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.11 (GNU/Linux)

iQIcBAEBAgAGBQJPBRz7AAoJEBgfAeV6NQkP1UIQALFcO8DyZkecTK5pAIcGez7k
ewjGBoCfjfO2NlRROuQm5CteXiH3Te5G+5ebsdRmGWVcah8QzN4UjxpKcTQRPB9e
/ehVI5BiBJq8GlOnaSRZpzsYobwKH6Jy6haAr3kPFK1lOXXyHSiNnQbydGw9BFRI
fSr//DY86BUILE8sGJR6FA8Vzjiifcv6mmXkk3ICrT8z0qY7m/wFOYjgiSohvYpg
x5biG6TBwxfmXQOaITdO5rO8+4mtLnP//qN7E9zjTYj4Z4gBhdf6hPSuOqjh1s+6
/C6IehRChpx8gwpdhIlNf1coz/ZiggPiqdj75Tyqg88lEr66fVVB2d7PGObSyYSp
```

```
HJl8llrt8Gnk1UaZUS6/eCjnBniV/BLfZPVD2VFKH2Vvvty8sL+S8hCxsuLCjydh
skpshcjMVV9xPIEYzwSEaqBq0ZMdNFEPxJzC0XISlWSfxROm85r3NYvbrx9lwVbP
mUpLKFn8ZcMbf7UX18frgOtujmqqUvDQ2dQhmCUywPdtsKHFLc1xIqdrnRWUS3CD
eejUzGYDB5lSflujTjLPgGvtlCBW5ap00cfIHUZPOzmJWoEzgFgdNc9iIkcUUlke
e2WbYwCCuwSlLsdQRMA//PJN+a1h2ZMSzzMbZsr/YXQDUWvEaYI8MckmXEkZmDoA
RL0xkbHEFVGBmoMPVzeC
=fRcg
-----END PGP SIGNATURE-----
je@isis:~/3301/stage_2$ gpg --verify welcome.txt
gpg: Signature made Thu 05 Jan 2012 04:46:03 AM CET using RSA key ID 7A35090F
gpg: Good signature from "Cicada 3301 (845145127)"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 6D85 4CD7 9333 22A6 01C3  286D 181F 01E5 7A35 090F
je@isis:~/3301/stage_2$ outguess -r problems.jpg problems.txt
Reading problems.jpg....
Extracting usable bits:   256999 bits
Steg retrieve: seed: 194, len: 1041
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

The key has always been right in front of your eyes.

This isn't the quest for the Holy Grail.  Stop making
it more difficult than it is.

Good luck.

3301
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.11 (GNU/Linux)

iQIcBAEBAgAGBQJPCBl3AAoJEBgfAeV6NQkPo6EQAKghp7ZKYxmsYM96iNQu5GZV
fbjUHsEL164ZLctGkgZx2H1HyYFEc6FGvcfzqs43vV/IzN4mK0SMy2qFPfjuG2JJ
tv3x2QfHMM3M2+dwX30bUD12UorMZNrLo8HjTpanYD9hL8WglbSIBJhnLE5CPlUS
BZRSx0yh1U+wbnlTQBxQI0xLkPIz+xCMBwSKl5BaCb006z43/HJt7NwynqWXJmVV
KScmkpFC3ISEBcYKhHHWv1IPQnFqMdW4dExXdRqWuwCshXpGXwDoOXfKVp5NW7Ix
9kCyfC7XC4iWXymGgd+/h4ccFFVm+WWOczOq/zeME+0vJhJqvj+fN2MZtvckpZbc
CMfLjn1z4w4d7mkbEpVjgVIU8/+KClNFPSf4asqjBKdrcCEMAl80vZorElG6OVIH
aLV4XwqiSu0LEF1ESCqbxkEmqp7U7CHl2VW6qv0h0Gxy+/UT0W1NoLJTzLBFiOzy
QIqqpgVg0dAFs74SlIf3oUTxt6IUpQX5+uo8kszMHTJQRP7K22/A3cc/VS/2Ydg4
o6OfN54Wcq+8IMZxEx+vxtmRJCUROVpHTTQ5unmyG9zQATxn8byD9Us070FAg6/v
jGjo1VVUxn6HX9HKxdx4wYGMP5grmD8k4jQdF1Z7GtbcqzDsxP65XCaOYmray1Jy
FG5OlgFyOflmjBXHsNad
=SqLP
-----END PGP SIGNATURE-----
je@isis:~/3301/stage_2$ gpg --verify problems.txt
gpg: Signature made Sat 07 Jan 2012 11:07:51 AM CET using RSA key ID 7A35090F
gpg: Good signature from "Cicada 3301 (845145127)"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 6D85 4CD7 9333 22A6 01C3  286D 181F 01E5 7A35 090F
```

The messages verifies both our assumptions, since they are indeed signed using the key ID 7A35090F and since the second one specifically says that the key “has always been right in front of your eyes”. In other words, it is likely to consist of the numbers we discovered being encoded as characters in the title of the page. The first message also specifically states that all messages from now on will be signed using the PGP key with ID 7A35090F.

All that remains now is to figure out which encoding scheme has been used so that we can apply the key to the text. Since a shift cipher was used in the original image (although it was used as a decoy), perhaps the numbers are different shift values. In other words, for each line of text, shift/rotate the first letter ten steps in the alphabet, rotate the second letter two steps, the third letter 14 steps, and so on, to get the plaintext. Implementing this in C results in the following:

```
je@isis:~/3301/stage_2$ cat decipher.c
#include <stdio.h>
#include <ctype.h>

int main(void)
{
        unsigned char key[] = {
                10, 2, 14, 7, 19, 6, 18, 12,
                7, 8, 17, 0, 19, 7, 14, 18,
                14, 19, 13, 0, 1, 2, 0
        };
        int c, i = 0;

        while ((c = getchar()) != EOF) {
                if (isalpha(c)) {
                        int base, off;
                        if (isupper(c))
                                base = 'A';
                        else
                                base = 'a';

                        off = c - base - key[i++ % sizeof(key)];
                        if (off < 0)
                                off += 26;

                        c = base + off;
                } else if (c == '\n')
                        i = 0;

                putchar(c);
        }

        return 0;
}
je@isis:~/3301/stage_2$ gcc -o decipher decipher.c -O -Wall -ansi -pedantic
je@isis:~/3301/stage_2$ head -3 reddit.txt
Ukbn Txltbz nal hh Uoxelmgox wdvg Akw; hvu ogl rsm ar sbv ix jwz
mjotukj; mul nimo vaa prrf Qwkkb aak kau ww Ukpsf, ogq Kzpox vvl luf
yh Qsrjfa, hvu Ktp hzs lbn ph Kipsy; ttv Sdmehpfjsf tad igr
je@isis:~/3301/stage_2$ ./decipher &lt; reddit.txt | head -3
King Arthur was at Caerlleon upon Usk; and one day he sat in his
chamber; and with him were Owain the son of Urien, and Kynon the son
of Clydno, and Kai the son of Kyner; and Gwenhwyvar and her
```

The file “reddit.txt” consists of the lines posted to the reddit page so far, in the order that they have been posted. Note that this is not in the exact order that they are shown on the reddit page. As you can see, our assumption was correct and we can now decipher every line of text that has been posted, and try to apply the book code that we got in the message hidden in the original image.

Using a small bash script, we can apply the book code to the text from reddit to retreive yet another hidden message:

```
je@isis:~/3301/stage_2$ ./decipher < reddit.txt > reddit-deciphered.txt
je@isis:~/3301/stage_2$ cat reddit-decode.sh
#!/bin/bash

while read line; do
        row=`echo $line | cut -d: -f1`
        col=`echo $line | cut -d: -f2`
        head -n$row reddit-deciphered.txt | tail -n1 | head -${col}c | tail -1c
done < bookcode.txt
echo
je@isis:~/3301/stage_2$ ./reddit-decode.sh
Call us at us tele phone oumBer two one four thsee nine oi nine si  oh  ihht
```

Although we can easily see which phone number is being refered to, it's obvious that the output is a bit garbled. For the sake of completeness, let's look into what the cause might be. The first letter that is garbled is the "n" in number that has been turned into an "o", then the "r" in three which have been turned into an "s" and so on. The upper case "B" may have been intended though, although it seems a bit off. There is actually a lower case "b" on the same line that is used for encoding the upper case "B", but the upper case one comes first.

When looking at the line corresponding to the "n" turning into an "o" (line 26, column 65), we can see that there is actually an "n" right before the "o" at column 65 (from the name "Kynon"). Looking further down, at the line corresponding to the "r" turning into an "s" (line 48, column 43), we can see that the expected "r" is right before "s" on this line as well (from the word "daggers").

Another thing in common for these particular lines of text is that they include a period somewhere before the character that has been decoded incorrectly. If we assume that periods, which end sentences, should count as two characters instead of one when applying the book code we get this, which looks a bit neater:

```
je@isis:~/3301/stage_2$ perl -i -pne 's/\./. /g' reddit-deciphered.txt
je@isis:~/3301/stage_2$ ./reddit-decode.sh
Call us at us tele phone numBer two one four three nine oh nine six oh eight
```

So, to continue the challenge we need to call the (214) 390-9608, a Texas based phone number. Whoever is behind this challenge, they have obviously put some effort into it. :)

When calling the number, one is (or rather, was, the number has now been deactivated) greeted by the following message:
"Very good. You have done well. There are three prime numbers associated with the

original final.jpg image. 3301 is one of them. You will have to find the other two. Multiply all three of these numbers together and add a .com to find the next step. Good luck. Goodbye."

When examining the PGP key, we already noted that it included the number 845145127 in the description, and that this is the product of 3301, 503 and 509. When looking at the metadata for the original image, we also note this:

```
je@isis:~/3301/stage_1$ exiftool 3301.jpg | grep 50[39]
Image Width                     : 509
Image Height                    : 503
Image Size                      : 509x503
```

Seems like we've solved this stage as well, now let's head to http://845145127.com/ to find the next part of the challenge. :) When I first arrived at the http://845145127.com/ site, it just displayed an image of a cicada and a countdown. Using OutGuess again, the following signed message could be extracted from the cicada image:

```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

You have done well to come this far.

Patience is a virtue.

Check back at 17:00 on Monday, 9 January 2012 UTC.

3301
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.11 (GNU/Linux)

iQIcBAEBAgAGBQJPCKDUAAoJEBgfAeV6NQkPf9kP/19tbTFEy+ol/vaSJ97A549+
E713DyFAuxJMh2AY2y5ksiqDRJdACBdvVNJqlaKHKTfihiYW75VHb+RuAbMhM2nN
C78eh+xd6c4UCwpQ9vSU4i1Jzn6+T74pMKkhyssaHhQWfPs8K7eKQxOJzSjpDFCS
FG7oHx6doPEk/xgLaJRCt/IJjNCZ9l2kYinmOm7c0QdRqJ+VbV7Px41tP1dITQIH
/+JnETExUzWbE9fMf/eJl/zACF+gYii7d9ZdU8RHGi14jA2pRjc7SQArwqJOIyKQ
IFrW7zuicCYYT/GDmVSyILM03VXkNyAMBhG90edm17sxliyS0pA06MeOCjhDGUIw
QzBwsSZQJUsMJcXEUOpHPWrduP/zN5qHp/uUNNGj3vxLrnB+wcjhF8ZOiDF6zk7+
ZVkdjk8dAYQr62EsEpfxMT2dv5bJ0YBaQGZHyjTEYnkiukZiDfExQZM2/uqhYOj3
yK0J+kJNt7QvZQM2enMV7jbaLTfU3VZGqJ6TSPqsfeiuGyxtlGLgJvd6kmiZkBB8
Jj0Rgx/h9Tc4m9xnVQanaPqbGQN4vZF3kOp/jAN5YjsRfCDb7iGvuEcFh4oRgpaB
3D2/+Qo9i3+CdAq1LMeM4WgCcYj2K5mtL0QhpNoeJ/s0KzwnXA+mxBKoZ0S8dUX/
ZXCkbOLoMWCUfqBn8QkQ
=zn1y
-----END PGP SIGNATURE-----
```

Just like before, the message is signed using the Cicada 3301 key. The challenge so far have been a quite fun, and rather different, experience and I'm looking forward to see what comes next.

When the countdown was finished, at 17:00 UTC January 9 2012, it was replaced by strings of digits resembling GPS coordinates. Also, the image of the cicada now contained another signed text containing the same GPS coordinates as on the web page, except for two that were only on the webpage (37.577070, 126.813122 and 36.0665472222222, -94.1726416666667):

```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

52.216802, 21.018334
48.85057059876962, 2.406892329454422
48.85030144151387,2.407538741827011
47.664196,  -122.313301
47.637520, -122.346277
47.622993, -122.312576
37.5196666666667, 126.995
33.966808, -117.650488
29.909098706850486 -89.99312818050384
25.684702, -80.441289
21.584069, -158.104211
- -33.90281, 151.18421
3301
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.11 (GNU/Linux)

iQIcBAEBAgAGBQJPChn7AAoJEBgfAeV6NQkPZxMP/05D9TkSpwRaBXPqYthuyqxx
uo+ZDyr/yVIlAdurTBiWb3aGxKJjtWg/vlcHcatK0TGL2qaHwB/FFZQAaqOyU7Zf
DXdpWr8PWoWhpWNYUK8IrOaYu1SmWlJnkTdUSzGrX0lbwjwMmJJoPNS7CJuO6MaA
2GIwpv2G7lYqnH3xeX3kzGlPMsVb/wucKRjobsbdbreh1SNuQuRnhfe4s+oHTTqs
XjtGL/VhBI0DUAdfLqW7z4C+Gvbx6okC8x5Sj2N2UTJOiyMYXz5+QyHoA6fo9g5V
6zodNpx/RvxuZP2Ssc9TqERgTo5FjRBpON1vjDalHgg0H2Fus2LK3gh+NZfj1i5b
Oqa4Cqd9epI2pe+glXn86j9crS+2BEAr1cguqAFepvI9sdFEornDja4VXwDtUdM8
9hMVkU5NiTUYfvxZbL6W7rHIF7wxjGUwpe1ViuixG+cKNfv0enrt60PrtDByBOWI
9LLIUE0cB5HDT1xrczZ/55CtuM3Zf07/l0nLFdmgR0oa8KUA9gWcPs6S1EpBa185
VcyOTqbpIPiT8neiJEkXarbJeFk15m1P73Fr8XZxdj7EHK0aOwGYcc8e4PmW/dSh
gcrSNXiePCbcRVRD2n9L47C0LkNyRpoBkmjvtpcRyp5ISe+0xcx/QI+gc1lkSijC
89qV+ymCHae1RiSDxVbd
=ZJ37
-----END PGP SIGNATURE-----
```

Using Google Maps (maps.google.com) I could search for each of these locations, and in most cases even get a street view. The locations were spread out around the world without any obvious connection (USA, Poland, France, South Korea and Australia), except for perhaps each of them being home to some talented hackers. At this point I thought it would be the end of the game for me, since I am far away from all of these locations.

I was still very curious on how the challenge would continue though, and found that there are groups of people working on this from all over the world. One of these groups had set up an IRC channel at n0v4.com, and managed to get people to check out the locations at the specified GPS coordinates. What they found was notes attached to lightpoles, with the cicada image and a QR code. When scanning the QR code, they got image URLs with a black

and white image of a cicada and the text “everywhere” and “3301”. Each image also contained a hidden signed message. Even though there were 14 locations, only two different messages were used though.

One of them had with the following text at the top of the message (full message here):

```
In twenty-nine volumes, knowledge was once contained.
How many lines of the code remained when the Mabinogion paused?
Go that far in from the beginning and find my first name.
```

The other one had this text (full message here here):

```
A poem of fading death, named for a king
Meant to be read only once and vanish
Alas, it could not remain unseen.
```

They both also included a 22 line book code. Both of them included the text “the product of the first two primes” at line 3 and 15, and one of them also included the text “the first prime” at line 8. This probably means that the characters on these positions should be replaced with the numbers described. Note that the definition of a prime number is a natural number greater than 1, with no positive divisors other than 1 and itself. This means that the first two prime numbers are two and three.

The three lines of text in each message seemed likely to be a hint to which book/text to use as the key for the included book code. By googling for some keywords in the second message (poem fading death read only once vanish), the Wikipedia entry for a 300-line poem by William Gibson is among the first hits. The poem is called Agrippa (a book of the dead) and according to Wikipedia “Its principal notoriety arose from the fact that the poem, stored on a 3.5" floppy disk, was programmed to erase itself after a single use; similarly, the pages of the artist’s book were treated with photosensitive chemicals, effecting the gradual fading of the words and images from the book’s first exposure to light.”. This fits the description perfectly.

When googling for william gibson agrippa, the first hit is http://www.williamgibsonbooks.com/source/agrippa.asp. Taking this text, including line breaks, as the key for the book code results in the following:

```
je@isis:~/3301/stage_3$ cat agrippa-decode.sh
#!/bin/bash

while read line; do
        if [ "$line" = "the product of the first two primes" ]; then
```

```
                echo -n 6
        else
                row=`echo $line | cut -d: -f1`
                col=`echo $line | cut -d: -f2`
                head -n$row agrippa.txt | tail -n1 | head -${col}c | tail -1c
        fi
done < agrippa-code.txt
echo
je@isis:~/3301/stage_3$ ./agrippa-decode.sh
sq6wmgv2zcsrix6t.onion
```

Judging by the ".onion" at the end of the string, this is actually an anonymous hidden service in the Tor network. Unfortunately, by the time I arrived at this stage the Tor service was not available anymore. 3301 had concluded the last couple of messages with "You've shared too much to this point. We want the best, not the followers. Thus, the first few there will receive the prize.", so it was probably first come first served. The ones who were lucky enough to arrive in time (most of which did not solve much or any of this challenge themselves, since people were sharing their solutions) got to enter their e-mail addresses and were informed that they would be contacted in few days.

By this time, someone noticed that the DNS entry for 845145127.com had been removed. By using the IP (75.119.203.244) it was found that the page that recently had GPS coordinates had changed yet again, to a seemingly empty page. On a closer look it turned out to consist entirely of spaces, tabs and linebreaks. Since every line contained a multiple of eight spaces/tabs, it seemed likely to be a plain binary code. This was confirmed by:

```
je@isis:~/3301$ wget -q -O- http://75.119.203.244/ > 3301.html
je@isis:~/3301$ perl -pne 's/[^\s]//g;s/\t/0/g;s/ /1/g;s{([01]{8})}{chr(oct("0b$1"))}sgex

-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

162667212858
414974253863
598852142735
876873892385
935691396441
316744223127
427566844663
644169769482
889296759263
963846244281
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.11 (GNU/Linux)

iQIcBAEBAgAGBQJPDRkvAAoJEBgfAeV6NQkPVuMP/3ZyAgwsko/B2T9Ew1yqAKVy
K9//wIWCRvMyZ4k79ApqvOJAlezgHTsAM8XG/I71bAG+2wMOXNJfTj/SFONEEbS5
BOp9UP7LHn1j3NKoESrDzsKd+u3oHoNnhs628aLrc8uDqbn/6DNUnObu5Tn3unu0
zZ3NjSs/A5QQX8O56RsK81eSJ2fifbr4NYfHBeUTeVe17nsr48WQI7qc9UVWlPsM
91FWsvhX+WohX8DyFWJmtz0lLvmh3jN+oE8WFPTVbcVCM+eiDt0TqkUNlmq/fxbd
X2Sbs8zMxDXNQWrw58TcSC6oLfXSXZnjh8uTMwrQ0tNdRXHDndgPiurXz62XjVjf
4AhSXBoXF9CHTOyGGEqvfNvFMKyz968iMZDXDNBrM8pkxx1xBHhAnoEznVpeMhII
+IfBTnV8x9lSNgFhmham5eEZlWvqRidqes8EAriqGA6uZokCq7X1IeMHo52ACWmP
2bJsCV5wZDc52c3JnwKe+cAcbsA4OWCNEH29lAsgFw5079BP8lkpY3AH2+8kqs0X
QvqsaMuUq5ZHEaZMgdD0VKYlRrKdhOiDjtJVxoXk1b7YBOV8dZZBXJbEIbTvaof4
```

```
yhgUObovx/VFGmsenp+j3nBCxgEO22SgNW3B3pN0yuIMCqccWEZ1nME/QOwLa85n
HOmElIXvK13Q9m545RtP
=Q1Fy
-----END PGP SIGNATURE-----
```

The message simply contains ten different 12 digit numbers. As it turns out, each of these correspond to image URLs such as: http://75.119.203.244/NUMBER.jpg

Each of these images contains a hidden message that can be extracted with outguess, and it turns out that it's the same messages that could be extracted from the images found through QR codes on notes at the GPS-coordinates mentioned earlier. Turns out we didn't have to be at one of those locations after all. :)

Regarding the remaining code, it is very likely to refer to the same .onion site as before. Just to be sure, and not to leave out any piece of the puzzle, it would be nice to solve that one too though.

My thoughts so far are these:

"In twenty-nine volumes, knowledge was once contained" may refer to the 11th edition of Encyclopedia Britannica, which consisted of exactly 29 volumes and that is now in the public domain and available for download since it was released back in 1910-1911.

Regarding "How many lines of the code remained when the Mabinogion paused?", note that the text posted to the reddit page is from "The Lady of the Fountain", which is the first out of eleven stories from medieval Welsh manuscripts in the collection called the Mabinogion. Also note that there was a pause for about 24 hours after the 65:th encoded line of text was posted to the reddit page. After that, new encoded lines have been posted about every 6th or 7th hour.

Assuming the code will continue until "The Lady of the Fountain" is finished, we will need to figure out the total number of lines in that story. To do that, we need to find the text that 3301 uses as their source, so that line breaks are placed on the same positions. After a bit of searching around it turns out that the source that 3301 uses is from Project Gutenberg (here). Blank lines are discarded, and lines with only one word on them are being appended to the preceding line. Applying those rules to the entire text of "The Lady of the Fountain" results in a total of 833 lines. Thus, the number of lines of code that remained when the Mabinogion paused is 833 – 65 = 768 (which also happens to be 512+256, but I guess that may be a mere coincidence after all).

Finally we have “Go that far in from the beginning and find my first name”, which could mean a number of things. My guess is that we should go 768 words, sentences, word definitions, characters or pages into the 11th edition of Encyclopedia Britannica. Question is where we are supposed to go from there, since it ends with “and find my first name”. By this, I assume we should only find a certain name at this particular position, and then from this name find the actual text to use as the key for the book code.

I also noticed that the code for this part only use 27 lines, with columns ranging from 1-66 and many columns being above 30-40. This rules out most poems, that usually don’t have long lines. It could very well be a text straight from the Encyclopedia Britannica, however. Due to the large number of possibilities I have not looked into it much further than this, and so far I don’t think anyone have come up with the solution for this particular puzzle. So, anyone up for it? :)