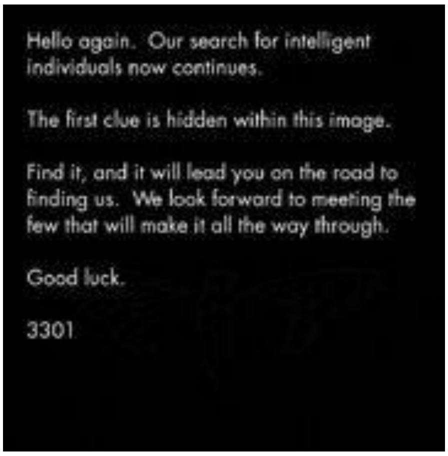
What Happened Part 1 (2013)

The Invitation



Hello again. Our search for intelligent individuals now continues.

The first clue is hidden within this image.

Find it, and it will lead you on the road to finding us. We look forward to meeting the few that will make it all the way through.

Good luck.

3301

The original image uploaded on January 4th.

It had been exactly 366 days since the 2012 Cicada puzzle began. Nothing had happened in 11 months.

Until the 5th of January 2013, when a second image was posted to /x/ and /b/ imageboards on 4chan. There were three threads where 3301 posted 232.jpg on 4chan, two times on /b/ 23 hours apart(1,2) and once on /x/.

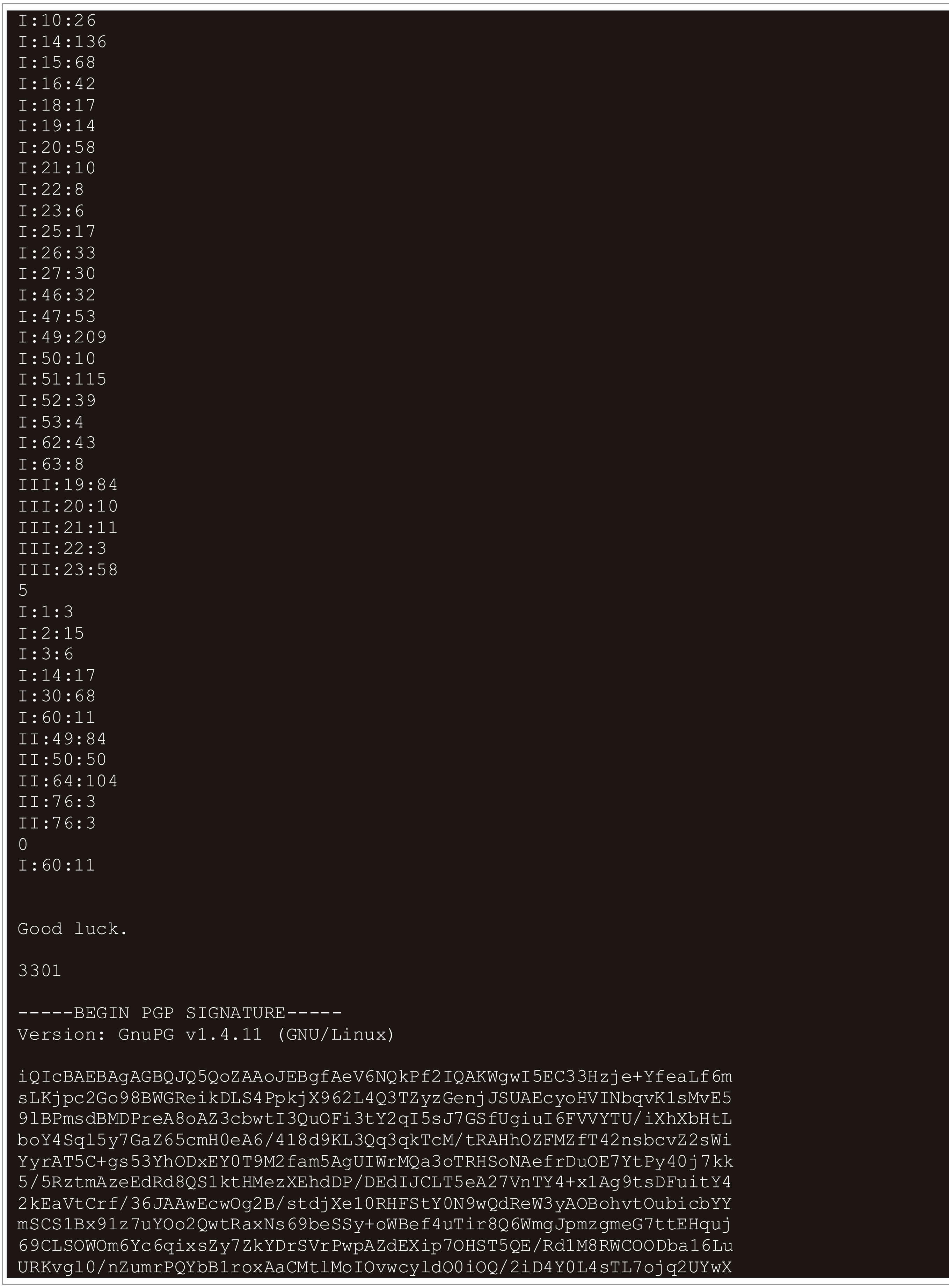
There was another thread one day earlier, /b/ thread on 4th that mentioned "warning pastebin" and SMS4TOR (onion.to link) service.

```
Hello again. Our search for intelligent
individuals now continues.

The first clue is hidden within this image.

Find it, and it will lead you on the road to
finding us. We look forward to meeting the
few that will make it all the way through.

Good luck

3301
```

The image was processed by the steganographic tool *outguess*. This message was the result. A more analytic look reveals the use of a book cipher. To decrypt the message, one needs to find the text that was used for encrypting.

[Collapse] PGP-Signed Message

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

Welcome again.

Here is a book code.  To find the book, break this riddle:

A book whose study is forbidden
Once dictated to a beast;
To be read once and then destroyed
Or you shall have no peace.


I:1:6
I:2:15
I:3:26
I:5:4
I:6:15
```



```
I:10:26
I:14:136
I:15:68
I:16:42
I:18:17
I:19:14
I:20:58
I:21:10
I:22:8
I:23:6
I:25:17
I:26:33
I:27:30
I:46:32
I:47:53
I:49:209
I:50:10
I:51:115
I:52:39
I:53:4
I:62:43
I:63:8
III:19:84
III:20:10
III:21:11
III:22:3
III:23:58
5
I:1:3
I:2:15
I:3:6
I:14:17
I:30:68
I:60:11
II:49:84
II:50:50
II:64:104
II:76:3
II:76:3
0
I:60:11


Good luck.

3301

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

```
iQIcBAEBAgAGBQJQ5QoZAAoJEBgfAeV6NQkPf2IQAKWgwI5EC33Hzje+YfeaLf6m
sLKjpc2Go98BWGReikDLS4PpkjX962L4Q3TZyzGenjJSUAEcyoHVINbqvK1sMvE5
9lBPmsdBMDPreA8oAZ3cbwtI3QuOFi3tY2qI5sJ7GSfUgiuI6FVVYTU/iXhXbHtL
boY4Sql5y7GaZ65cmH0eA6/418d9KL3Qq3qkTcM/tRAHhOZFMZfT42nsbcvZ2sWi
YyrAT5C+gs53YhODxEY0T9M2fam5AgUIWrMQa3oTRHSoNAefrDuOE7YtPy40j7kk
5/5RztmAzeEdRd8QS1ktHMezXEhdDP/DEdIJCLT5eA27VnTY4+x1Ag9tsDFuitY4
2kEaVtCrf/36JAAwEcwOg2B/stdjXe10RHFStY0N9wQdReW3yAOBohvtOubicbYY
mSCS1Bx91z7uYOo2QwtRaxNs69beSSy+oWBef4uTir8Q6WmgJpmzgmeG7ttEHquj
69CLSOWOm6Yc6qixsZy7ZkYDrSVrPwpAZdEXip7OHST5QE/Rd1M8RWCOODba16Lu
URKvgl0/nZumrPQYbB1roxAaCMtlMoIOvwcyldO0iOQ/2iD4Y0L4sTL7ojq2UYwX
```

```
bCotrhYv1srzBIOh+8vuBhV9ROnf/gab4tJII063EmztkBJ+HLfst0qZFAPHQG22
41kaNgYIYeikTrweFqSK
=Ybd6
-----END PGP SIGNATURE-----
```

Check Signature

This poem, introducing the secret message, was a nudge towards the right text. After a bit of debate, the text that was used to encrypt the book cipher was discovered.

The Law *(Liber AL vel Legis)*

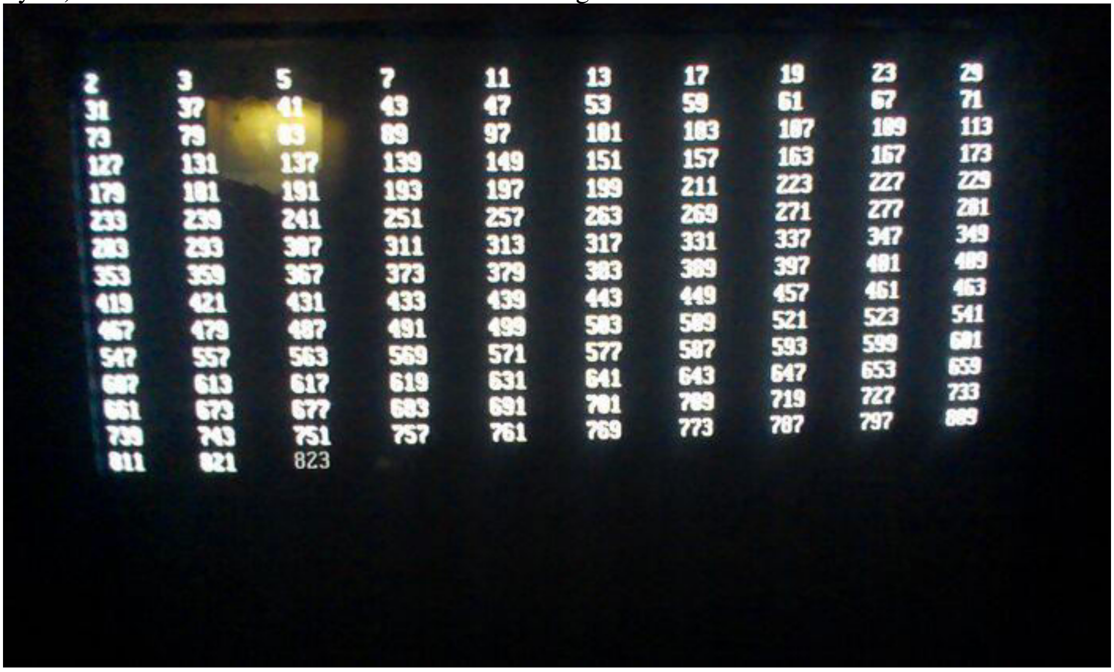
The book that was used to hide the message was *Liber AL vel Legis* by Aleister Crowley. Also known as "The Book of Law ", it is available online, and can be found here . The first line *I:1:6* points toward the 6th character of the first line in the first chapter, an 'h' in this case. It was assumed that spaces weren't counted. Punctuation, however, influenced the character chosen for the plaintext. During decrypting, we found that dashes were vital to the process, so we kept them in the plaintext. Using these rules, we encrypted the book cipher and came up with the decrypted message.

```
https:--www.dropbox.com-s-r7sgeb5dtmzj14s-3301
```

We agreed upon substituting the dashes with slashes and came up with a hyperlink.

The System

The hyperlink (mirror, mirror 2) directed to a dropbox address with a file of 130MB ready for download. After downloading, the file was analysed and a quick check for magic bytes (header bytes) revealed that the offered file was an .iso image.



```
2    3    5    7    11   13   17   19   23   29
31   37   41   43   47   53   59   61   67   71
73   79   83   89   97   101  103  107  109  113
127  131  137  139  149  151  157  163  167  173
179  181  191  193  197  199  211  223  227  229
233  239  241  251  257  263  269  271  277  281
283  293  307  311  313  317  331  337  347  349
353  359  367  373  379  383  389  397  401  409
419  421  431  433  439  443  449  457  461  463
467  479  487  491  499  503  509  521  523  541
547  557  563  569  571  577  587  593  599  601
607  613  617  619  631  641  643  647  653  659
661  673  677  683  691  701  709  719  727  733
739  743  751  757  761  769  773  787  797  809
811  821  823
```

Screenshot from the video showing the prime number sequence being printed.

The image file was downloaded by multiple solvers and either burned to disc to run on a computer or opened in a virtual drive. Looking into its contents, we find three directories, "data", "boot" and "audio".

When booting from the image, a boot sequence appeared, printing a sequence of numbers to the screen. Investigating the sequence revealed that the live image prints out all prime numbers up to 3301. There were temporary two-second pauses at 1033 and 3301, where it stops at the latter and moves to the second stage. The next, and last stage of the procedure is a screen that reads:

```
@1231507051321

The key is all around you.

Good luck.

3301
```

full

Further analysis of the live image turned up the routine responsible for the display of the prime numbers. It is a linux shell script (found here , for those interested), which, luckily, is human-readable. It does not calculate prime numbers, like some suggested, but connected the printing command with a sleep command. In most cases, the sleep time is 0.5 seconds. In case of the primes 1033 and 3301 however, the sleep time is 2 seconds, which manifested the relevance of those two numbers. Also, this clue said "you" not "we", differing from the last one in the choice of words.
Also found in the image was this PGP signature, which has been verified to be 3301's official signature released during the puzzle in 2012.

It is possible to interrupt the boot sequence by pressing CTRL+C. User "tc" is active and does not require a password, is in sudoers file with no prompt. `sudo ash` to raise to root. Further inspection revealed nothing that is not listed in this wiki.

Another video of Cicada OS: http://www.youtube.com/watch?v=PTFa6dwCEiA

The Music

The folder "audio" contained an audio recording. The title of the recording was "761.mp3" and can be downloaded here. The ID3 tags show us that the title of the file is "The Instar Emergence" and the artist "3301". The used instrument is a guitar, with distorting effects on it. On the track, a reversed guitar is played and amplified throughout. The song has been deconstructed and checked for hidden reversed messages, but as of yet has turned up nothing out of the ordinary.

The song is in the key of D♭ minor with a custom guitar tuning of D♭-A♭-D♭-G♭-A♭-D♭

You can listen to the song here:

The song found on the ISO

Key points about the track is the initial 'breath' sound, believed to be the sound of many cicadas and the tempo changes, beginning at approximately 135 bpm, accelerating to 145bpm, then slowing to 125bpm. This has led some to believe that the song has been slowed down by 5%. The only instruments used were a guitar acoustic and electric and an effect driven bass drum. A draft spectral anlysis shows a constant hum at 15.4-16.1kHz, and empty notches under 500Hz starting from 1:56. A hexdump of the mp3 file revealed the following message:

```
The Instar Emergence

Parable 1,595,277,641
Like the instar, tunneling to the surface
We must shed our own circumferences;
Find the divinity within and emerge.
```

The original message had "\n" attached to the end of each line. This character sequence is used to indicate a new line in some programming languages. These were omitted due to the availability of proper formatting techniques.

The subgroup who were assigned the task of analyzing the poem/riddle above have speculated that circumferences might be a reference to perceived limitations rather than actual physical walls. "Find the divinity within and emerge" is most likely a reference to the divine ratio, or phi. Such shedding may also be a reference to the way Cicadas shed their shells.

It has also been pointed out that the song is 2:47 long, or 167 seconds, which is prime. It is also a reversal of the name of the file: 761.mp3, and 761 is also prime.

Meaning of Parable 1.595.277.641

The Twitter

While people still searched through the image to find more hints that may have been overlooked, somebody in the IRC found a twitter account which got our attention, to say the least. Multiple things were strange about that twitter. It fits the overall "style" of cicada, it was registered shortly after the first downloads of the live image and it had no followers. It was later found to be the reference on the boot CD to @1231507051321 (note: 1231507051321 is a palindromic prime number). The most striking thing about it though was the messages it tweeted.

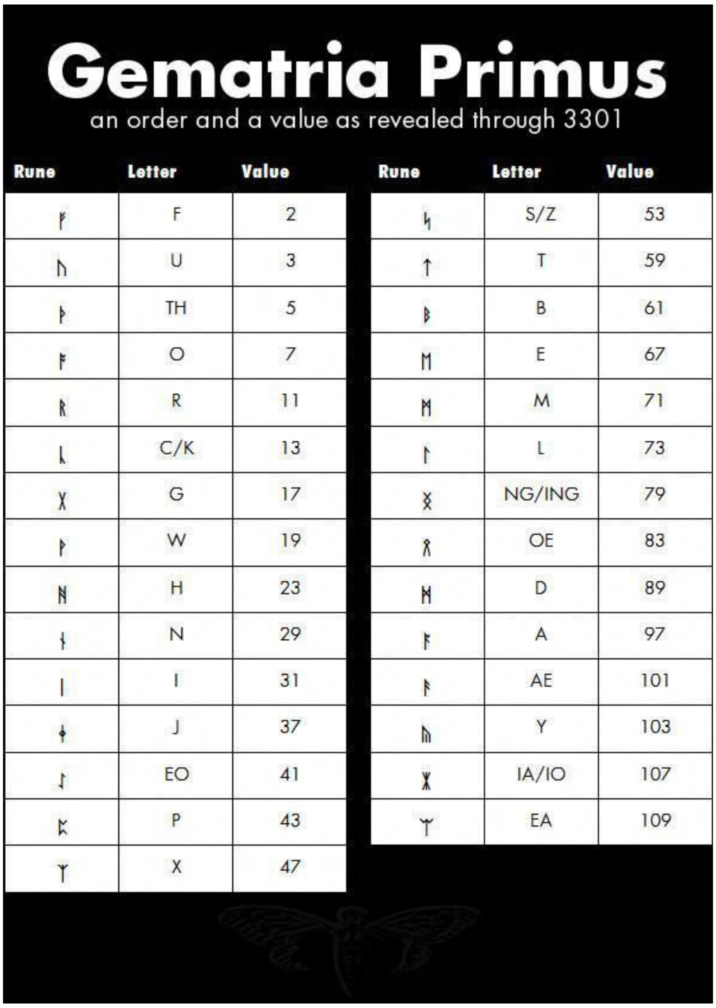
Each tweet consisted of an offset, and 65 bytes of hex code. For example, the first message went like so:

```
0000000:
b69ccce300104a464802545959580001008d0000ff8b6131616a6a632737293d3e322
b3b3e3f263a203c0c4762677c326767713d73716d697b6e3000505b494e47
```

3301 appears to have used a bot to post the tweets at 5 minute intervals (up until 0:00 GMT Jan 7), then onto four minute intervals until 19:00 GMT Jan 7, where it was seemingly random up until 22:04 GMT Jan 7, where it moved onto two minute intervals. The twitter bot stopped posting tweets at 4:52 GMT on Jan 8.

The meaning of the tweets and the rest of the files left the solvers stumped for several hours. A full feed of the tweets is avaliable here.

The Gematria



Gematria Primus

an order and a value as revealed through 3301

Rune	Letter	Value	Rune	Letter	Value
ᚠ	F	2	ᛋ	S/Z	53
ᚢ	U	3	ᛏ	T	59
ᚦ	TH	5	ᛒ	B	61
ᚩ	O	7	ᛖ	E	67
ᚱ	R	11	ᛗ	M	71
ᚳ	C/K	13	ᛚ	L	73
ᚷ	G	17	ᛝ	NG/ING	79
ᚹ	W	19	ᛟ	OE	83
ᚻ	H	23	ᛞ	D	89
ᚾ	N	29	ᚪ	A	97
ᛁ	I	31	ᚫ	AE	101
ᛄ	J	37	ᚣ	Y	103
ᛇ	EO	41	ᛡ	IA/IO	107
ᛈ	P	43	ᛠ	EA	109
ᛉ	X	47			

Result of xor-ing the twitter information with 761.mp3

After a day of fruitless searching, an IRC user did the impossible and solved the next puzzle. This user took the 761.mp3 file, and XORed it with the file produced by following the instructions in the twitter. The result was a .jpg file. It was possible to "pre"-construct the image resulting from the tweets.

The .jpg file appears to be a rune table, consisting of three columns, named "Rune", "Letter" and "Value", and 29 entries. "Rune" contains the actual rune character, "Letter" contains one or more plain text characters and "Value" contains a number. It is interesting to note that the numbers to be found in "Value" are all ascending primes, building the sequences of the first 29 prime numbers. As a member in the IRC pointed out, the runes stem from the Anglo-Saxon rune set, and the letters are in the order of the Anglo-Saxon runes. It was revealed that this is a fully-blown Gematria, which can be applied to different pieces of text to reveal interesting numbers. 'The Instar Emergence', for example, produces 761, which is the name of the file and the file's time signature reversed.

It was soon discovered that this image, like the very first one, contained a hidden message, once again masked via OutGuess. The message itself can be found here. As in every message from Cicada, the content was followed by a PGP signature, which proved the authenticity of the message

[Collaps

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

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

iQIcBAEBAgAGBQJQ5lDTAAoJEBgfAeV6NQkP7nMQAJVg7DQiIA7NpkacR0RA4eBs
NZHJBQNHO2P22h+aFfP/rI1gjGaV3hMWaa2sQ4Vbi/W8eZuH40AsmZUy3EOb+4j0
3cJRJgAJI99ZjDcVXITm5VyUv+WIqCzBr+bHMK7pkMYQ/rEzeWD56tlsrDgFdjmh
PA/b7XrDcofd9JfBNFI7D/sF84HL2ig5baNo+MGjYl4Dq2cHX+SAafXmlN9PXFjx
HRBbuoMLlviKywQ8MnePBPYG6V8sIMmrJlHS5ZcNEaSJ9nGL4X0XbECqV79ermye
1EeNKcckoeeZMU86SabfMeyZozG04Vkbemn8JH5cssbuF8hf4fdN/LSP4NG0r5y9
jfRv7z59pL577ZpGAju5zBtlCBUvmxxNYR5IGLg+Fi/ICqcRC98mzesFnQ7wbDLS
HKyV95SBQK82bbqSREBfIrrNb+MjVtJwIvOY5OPTBViHPqrIuMw8KDGfSvw9ncCt
dase7vUjXxIrn36xDSRN6cMzTmFZ9lkQYkRAYq5ApERud+JfKCwszG/UxRwo1WOU
0ALaWXq5VMp+w5pvQkqg9eHpOriG9Z11VLdb53eTmxKrwyX/2eaiybsnMrRNuxv1
iE8PVRkifCcJccw1bGq8TyCQF3a5ozeiBRngAUT7BwZhLa4bShtki7amR0ZZgbKk
8JRMGvoSA5NNTEwvUhwl
=ZeNf
-----END PGP SIGNATURE-----
```

The Onion, part 1 of 2

After finally getting a message from 3301, the solvers found that it was, to quote the IRC, 'Mostly Blank'. The message, it turned out, contained a mixture of tabs and spaces. The solvers converted this to binary, then again to ASCII, then they found the next message:

```
"Come to emiwp4muu2ktwknf.onion"

"We shall await you there."

"Good luck."

"3301"
```

A quick filler: emiwp4muu2ktwknf.onion is a website that can only be accessed through Tor, which is using a hidden service URL, similar to the last Cicada puzzle.

Upon visiting the website, the solvers were presented with the following message:

Note: the formatting may be a little off on your screen. Press control/command and - to view the full message.

```
Web browsers are useless here.


      ,+++77777++=:,                    +=
,,++=7++=,,
    7~?7   +7I77 :,I777  I          77 7+77 7:
,?777777??~,=+=~I7?,=77 I
=7I7I~7  ,77: ++:~+7 77=7777 7     +77=7 =7I7     ,I777= 77,:~7 +?7,
~7   ~ 777?
77+7I 777~,,=7~  ,::7=7: 7 77   77: 7 7 +77,7 I777~+777I=   =:,77,77
77 7,777,
  = 7  ?7 , 7~,~  + 77 ?: :?777 +~77 77? I7777I7I7 777+77   =:, ?7
+7 777?
      77 ~I == ~77= +777 777~: I,+77?  7  7:?7? ?7 7 7 77 ~I   7I,,?7
I77~
       I 7=77~+77+?=:I+~77?     , I 7? 77 7   777~ +7 I+?7
+7~?777,77I
         =77 77= +7 7777         ,7 7?7:,??7     +7    7   77??+
7777,
             =I, I 7+:77?         +7I7?7777 :             :7 7
                7I7I?77 ~         +7:77,     ~         +7,::7   7
               ,7~77?7? ?:         7+:77777,           77 :7777=
                ?77 +I7+,7         7~  7,+7  ,?       ?7?~?777:
                   I777=7777 ~     77 :  77 =7+,    I77  777
                     +      ~?     , + 7    ,, ~I,  = ? ,
                                    77:I+
                                    ,7
                                     :77
                                        :

Welcome.
```

Establishing a connection

The solvers soon found that web browsers were indeed useless, and that we would have to telnet into the website through the tor network. Some solvers did so, and they found that the website included an interactive shell. They could type in any number to have it factorized, 'count' to have it count up prime numbers, 'quit' to quit, and 'hello' to return a message. Please see here for the original message.

It was soon discovered that the messages could be turned into ASCII which created another message, again GPG signed by 3301. The message reads as follows:

```
Very good.

You have done well to come this far.

xsxnaksict6egxkq.onion

Good luck.

3301
```

This led us to another, the second .onion address.

The Clues

As the solvers patiently waited for more news about the second .onion, they continued to explore other options that they may have overlooked in the blind rush towards victory.

A new message was found by telnetting "hint" or "clue". And XORing result with _560.00 file from the DATA folder on 3301.txt CD image.

```
You can't see the forest when you're looking at the trees.

Good luck.

3301
```

Full Message:
https://pastee.org/2zae9
"hint" output:
https://pastee.org/tjdbs

THE DIFFERENCE

In Cicada OS the solvers found two files named Wisdom and Folly in ./tmp

http://codeseekah.com/cicada/folly
http://codeseekah.com/cicada/wisdom
Wisdom is exactly the same as folly, but appears to represent no file type.

THE PRIMES

Telnetting 'primes' into the shell printed out a list of primes similar to the one on Cicada OS, but some primes were missing and two have extra spaces in front of them.

There were extra spaces between 29-31 and 3257-3259
And some missing primes between 71-1229

The missing primes are as follows:

```
   73     79     83     89     97    101    103    107    109    113
   127    131    137    139    149    151    157    163    167    173
   179    181    191    193    197    199    211    223    227    229
   233    239    241    251    257    263    269    271    277    281
   283    293    307    311    313    317    331    337    347    349
   353    359    367    373    379    383    389    397    401    409
   419    421    431    433    439    443    449    457    461    463
   467    479    487    491    499    503    509    521    523    541
   547    557    563    569    571    577    587    593    599    601
   607    613    617    619    631    641    643    647    653    659
   661    673    677    683    691    701    709    719    727    733
   739    743    751    757    761    769    773    787    797    809
   811    821    823    827    829    839    853    857    859    863
   877    881    883    887    907    911    919    929    937    941
   947    953    967    971    977    983    991    997    1009
1013
```

```
   1019   1021   1031   1033   1039   1049   1051   1061   1063
1069
   1087   1091   1093   1097   1103   1109   1117   1123   1129
1151
   1153   1163   1171   1181   1187   1193   1201   1213   1217
1223
```

You can use a hosted telnet service to access the tor website here.

The Onion, Part 2 of 2

Once the solvers had found the second .onion, the next logical step was to visit it with a browser. Upon arrival, they found the following:

```
Patience is a virtue.
```

Rummaging through the source code for the html, they found the following:

```
<html>
        <head><title>3301</title></head>
        <body>
                Patience is a virtue.
                <!-- which means, come back soon. -->
        </body>
</html>
```

Soon afterwards, someone attempted to telnet into it, producing an error message which contained the address of the VPS on which the site was hosted. Promptly afterwards, the site was taken down.

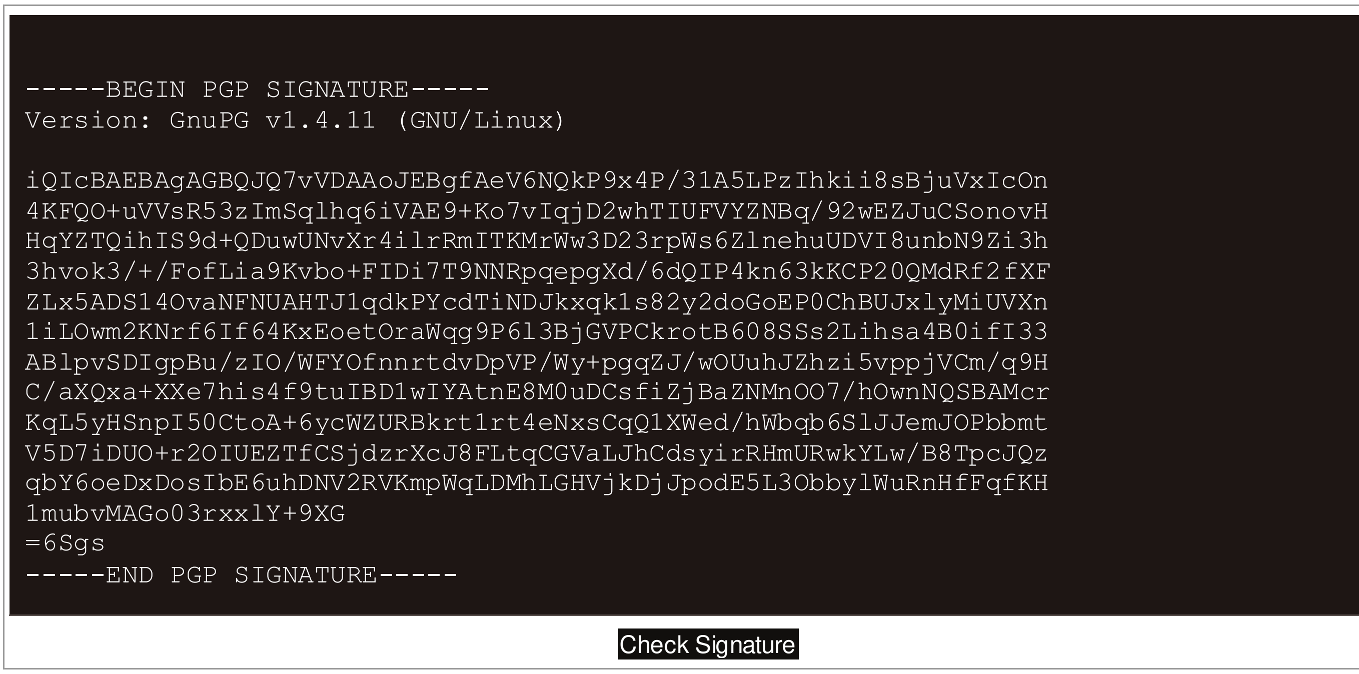
After this the 2nd Onion site finally reopened. The solvers got the following hint:

```
You already have everything you need to continue.
Sometimes one must "knock on the sky and listen to the sound."
Good luck.
```

This hint told the solvers that they needed to ping the website's IP address and listen to the reply. Each ping reply was laced with data bytes, which could be combined to make the following:

[Collapse] PGP-Signed Message

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


Well done.  You have come far.

pklmx2eeh6fjt7zf.onion

Good luck.

3301
```

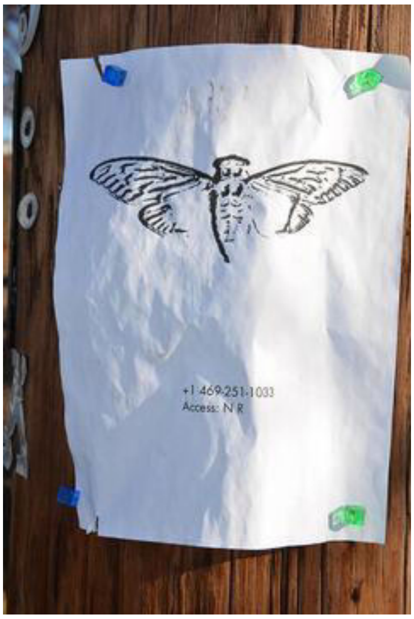


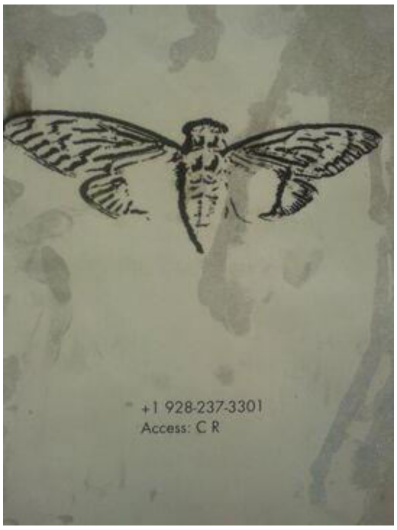
```
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.11 (GNU/Linux)

iQIcBAEBAgAGBQJQ7vVDAAoJEBgfAeV6NQkP9x4P/31A5LPzIhkii8sBjuVxIcOn
4KFQO+uVVsR53zImSqlhq6iVAE9+Ko7vIqjD2whTIUFVYZNBq/92wEZJuCSonovH
HqYZTQihIS9d+QDuwUNvXr4ilrRmITKMrWw3D23rpWs6ZlnehuUDVI8unbN9Zi3h
3hvok3/+/FofLia9Kvbo+FIDi7T9NNRpqepgXd/6dQIP4kn63kKCP20QMdRf2fXF
ZLx5ADS14OvaNFNUAHTJ1qdkPYcdTiNDJkxqk1s82y2doGoEP0ChBUJxlyMiUVXn
1iLOwm2KNrf6If64KxEoetOraWqg9P6l3BjGVPCkrotB608SSs2Lihsa4B0ifI33
ABlpvSDIgpBu/zIO/WFYOfnnrtdvDpVP/Wy+pgqZJ/wOUuhJZhzi5vppjVCm/q9H
C/aXQxa+XXe7his4f9tuIBD1wIYAtnE8M0uDCsfiZjBaZNMnOO7/hOwnNQSBAMcr
KqL5yHSnpI50CtoA+6ycWZURBkrt1rt4eNxsCqQ1XWed/hWbqb6SlJJemJOPbbmt
V5D7iDUO+r2OIUEZTfCSjdzrXcJ8FLtqCGVaLJhCdsyirRHmURwkYLw/B8TpcJQz
qbY6oeDxDosIbE6uhDNV2RVKmpWqLDMhLGHVjkDjJpodE5L3ObbylWuRnHfFqfKH
1mubvMAGo03rxxlY+9XG
=6Sgs
-----END PGP SIGNATURE-----
```

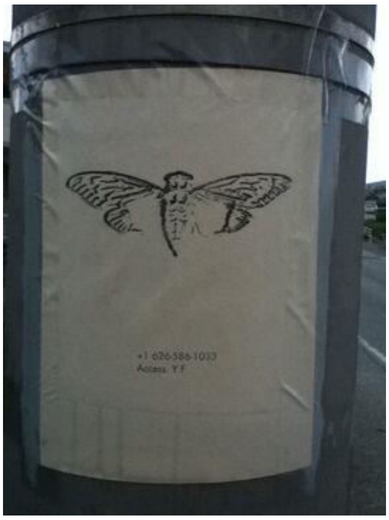
The Third Onion

On the third Onion page the solvers recieved a message instructing them to 'standby for coordinates'. They prepared to visit the addresses which these would undoubtedly lead them to.

Each poster had a phone number on it as well as an access code. Note that each phone number either ends in 3301 or 1033.







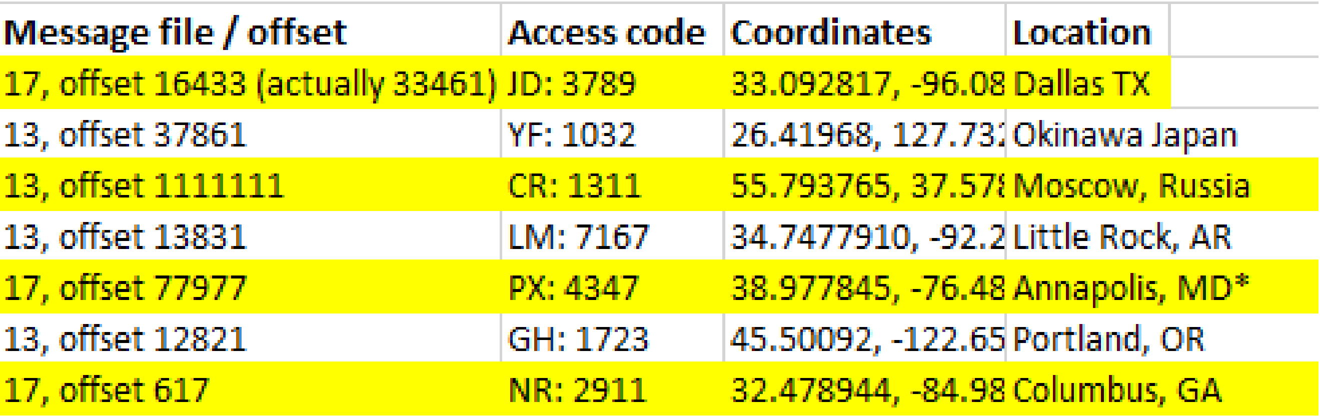
Calling the phone number gave an automated speech asking for a code to be typed into the dialer. Solvers soon realised that they had to convert the access code given in the poster to it's gematrified format and type that in. Upon doing so the following message was given (it varied depending on location, this one was for Portland):

```
Dataset:13
Offset:12821

Data:28C07E1B102D4D5C4C1A376E064477E1416FCC94928765
```

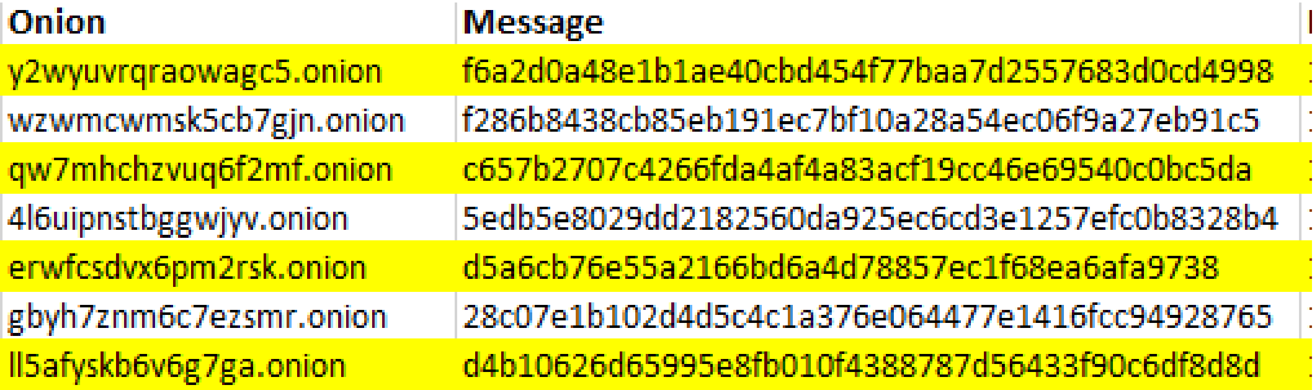
The data, when XORed with the 560.13 (the 13 coming from the dataset) file from DATA, provided the user with a string of text, notably in this case "gbyh7znm6c7ezsmr.onion". It's important to note that each location gave a different onion address.

All in all, 6 of the locations had their codes recovered, while the seventh was not physically visited, but the phone number obtained by wardialing all numbers ending in 1033.

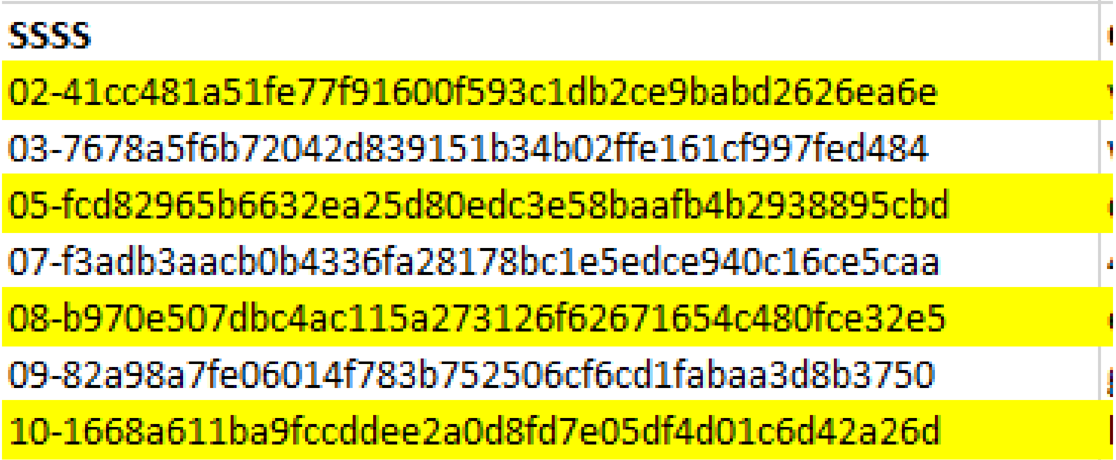
On each of these onion addresses (as listed in the table below), each solver was given an SSSS code, which stands for Shamir's Secret Sharing Scheme. A secret sharing scheme allows someone to share a secret with a certain number of people, who each get their own string. Once enough of these secrets come together, they can be combined to create the final secret. Each location, its SSSS code and some other data on each part is in the below table:



Message file / offset	Access code	Coordinates	Location
17, offset 16433 (actually 33461)	JD: 3789	33.092817, -96.08	Dallas TX
13, offset 37861	YF: 1032	26.41968, 127.73	Okinawa Japan
13, offset 1111111	CR: 1311	55.793765, 37.57	Moscow, Russia
13, offset 13831	LM: 7167	34.7477910, -92.2	Little Rock, AR
17, offset 77977	PX: 4347	38.977845, -76.48	Annapolis, MD*
13, offset 12821	GH: 1723	45.50092, -122.65	Portland, OR
17, offset 617	NR: 2911	32.478944, -84.98	Columbus, GA



Onion	Message
y2wyuvrqraowagc5.onion	f6a2d0a48e1b1ae40cbd454f77baa7d2557683d0cd4998
wzwmcwmsk5cb7gjn.onion	f286b8438cb85eb191ec7bf10a28a54ec06f9a27eb91c5
qw7mhchzvuq6f2mf.onion	c657b2707c4266fda4af4a83acf19cc46e69540c0bc5da
4l6uipnstbggwjyv.onion	5edb5e8029dd2182560da925ec6cd3e1257efc0b8328b4
erwfcsdvx6pm2rsk.onion	d5a6cb76e55a2166bd6a4d78857ec1f68ea6afa9738
gbyh7znm6c7ezsmr.onion	28c07e1b102d4d5c4c1a376e064477e1416fcc94928765
ll5afyskb6v6g7ga.onion	d4b10626d65995e8fb010f4388787d56433f90c6df8d8d



SSSS
02-41cc481a51fe77f91600f593c1db2ce9babd2626ea6e
03-7678a5f6b72042d839151b34b02ffe161cf997fed484
05-fcd82965b6632ea25d80edc3e58baafb4b2938895cbd
07-f3adb3aacb0b4336fa28178bc1e5edce940c16ce5caa
08-b970e507dbc4ac115a273126f62671654c480fce32e5
09-82a98a7fe06014f783b752506cf6cd1fabaa3d8b3750
10-1668a611ba9fccddee2a0d8fd7e05df4d01c6d42a26d

Once 5 of 10 SSSS codes had been retrieved, they could be decrypted to form their message, which was:

```
p7amjopgric7dfdi.onion
```

This was the fifth onion.

Here is how test looked: http://imgur.com/a/YFA1a

Here ends Part 1 of What Happened during the 2013 Cicada puzzle, and also ends the part of the 2013 puzzle that was fully publicly available. Part 2 is available here, and relies entirely on leaks (which may also comprise integrity).