

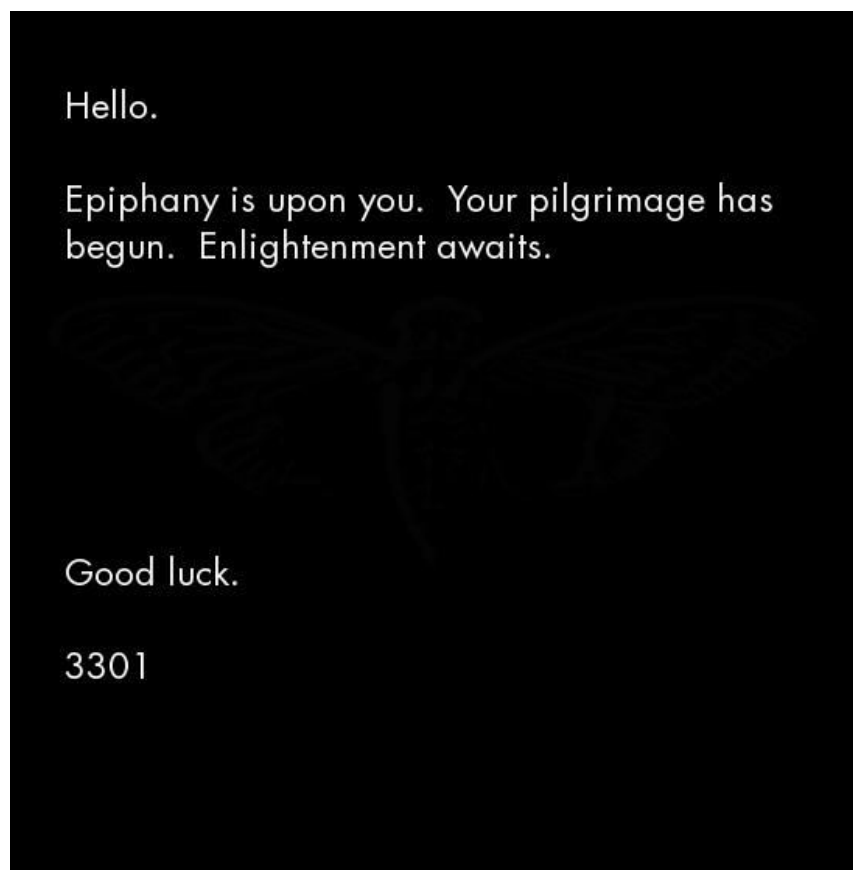
What Happened Part 1 (2014)

We (the authors) have chosen not to name any of the individuals responsible for specific solutions here, and instead refer to the solving community as a whole. Shoutouts to you few who actually solved the puzzle -- you know who you are.

The Beginning

The Twitter Image

In early January 2014, people gathered together on the net to wait for the next signs of life from Cicada 3301. After several fake puzzles, eventually a genuine message from Cicada was received. On January 6th, the [Twitter](#) account used by Cicada in 2013's puzzle was re-examined; after being inactive for about a year, it suddenly tweeted a link to an image on [imgur](#).



In line with earlier rounds, this image contained a steganographically hidden message recoverable using the program [outguess](#). Executing

```
outguess -r zN4h51m.jpg zN4h51m_output
```

yields

-----BEGIN PGP SIGNED MESSAGE-----

Hash: SHA1

The work of a private man
who wished to transcend,
He trusted himself,
to produce from within.

1:2:3:1
3:3:13:5
45:5:2:3
20:3:20:5
8:3:8:6
48:5:14:2
21:13:4:1
25:1:7:4
15:9:3:4
1:1:16:3
4:3:3:1
8:3:26:4
47:3:3:5
3
13:2:5:4
1:4:16:4

.
o
n
i
o
n

Good luck.

3301

-----BEGIN PGP SIGNATURE-----

Version: GnuPG v1.4.11 (GNU/Linux)

iQIcBAEBAGAGBQJSyJguAAoJEBgfAeV6NQkPsgAP/A3tMC3lpyFNAC/sj+Izu15S
CzUjZJMe20Gu9UMNokQ2UJabktv9w0GMyK17TrMkUcU+ZpjdzGNqKoE2ETVxLmD/
uBZtR5PnF9EE3D08tJUPN1vSrYNkYk+9zcaUJZMPNgYNct/CACutPwrOci9i9FDO
7BIpnhGqT3ZruqrSwO2Y73LJI1xxUt1XUqh1NQ+fJeAFMRkJBZzazkxRlgk3GGsF
fLrcEKrS+KBipV1EQaaKxjISc9hc2c1TfxE66evlkn+zLcoyDcYuyruNM5wiZzgM
2uR58c+XgWQgG5UuLFC1fvjDxUvDkrKt4mzEeaYSUm1MsYueuYklz4ydlg5Mf6l2
p1WyAxO52XfXVUZASk6VmaEQ0WjODTXvLeFTxUSDoKDMkvxDVxX6wGkufS9JwakB
nTZizZ8Ypv8GcNCuNNGd6gZ1Vk2MYntggXdX8INd0Itcd3QnLqbBnATDOinDxlOs
5zTrtyTHNaxxDagPfAbU1jMXM0aHd7PFAzjjp7kgCTWqMyBch+8Vt80bjkdL9iw8
Q3hxuanq8mh6nUGc+tNe0UfqKHEbE+jWIezYqgawJB0M9R5OhxWE+E+jPXtZKkXQ
JHYndPDrrsV8q27b7p0KN0+oblTkjqsItIAuLu7FNd0B4xb1jjp1Sbh7WJdZ/rbi
mCO0vN/obU9qK1Vfapy0
=6Gxk

-----END PGP SIGNATURE-----

This message is signed with the [PGP key](#) used by Cicada in past years, verifying its authenticity. This message marks the start of 2014's recruitment puzzle.

The Book Cipher

The message looked like a book cipher. Cicada has used book ciphers [in past years](#). Thanks to the hint in the beginning of the message the book was easy to find -- [Self-Reliance and Other Essays](#), by Ralph Waldo Emerson.

To illustrate how to solve this cipher, take the first line of the message:

```
1:2:3:1
```

This line references the first paragraph of the text, the second sentence in that paragraph, the third word in that sentence and finally the first letter in that word. In this case the first character of the third word of the second sentence of the first paragraph in *Self-Reliance and Other Essays* is 'a'. Following this scheme for the rest of the cipher (the solitary 3 points to the third character in "Ralph Waldo Emerson", 'l') and appending ".onion" to the result yields

```
auggnxjtvdbll3pv.onion
```

which is the address of a Tor hidden service. Visiting the service we found...

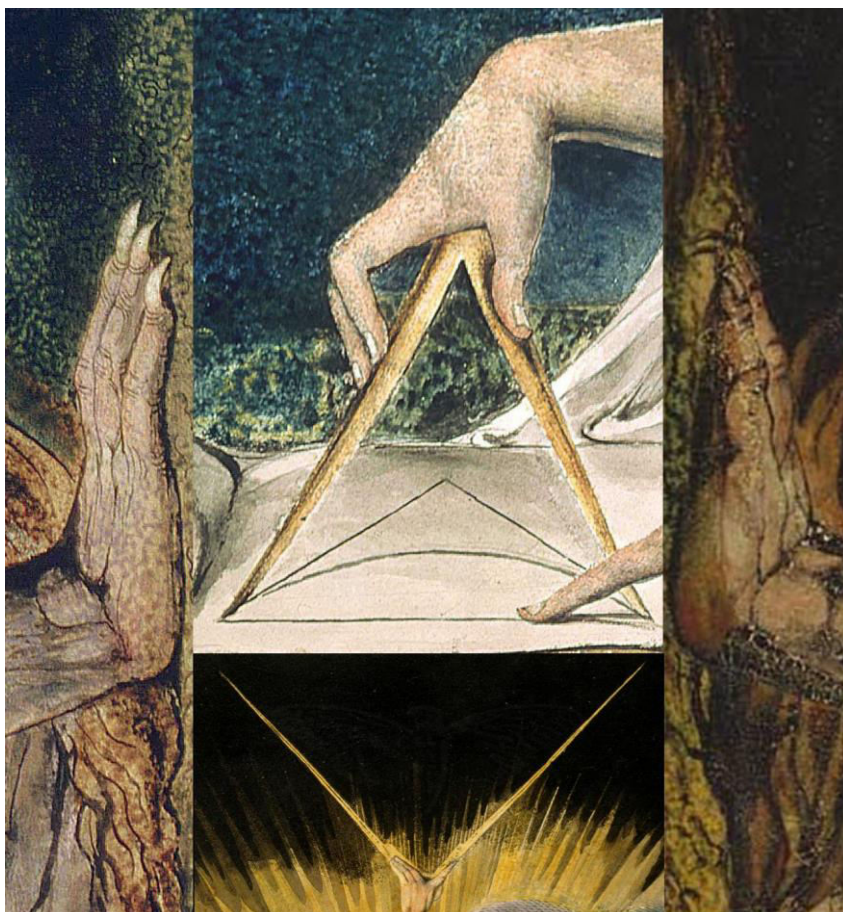
The First Onion

This onion has since gone offline.

The William Blake Collage

The hidden service featured [this image](#), which is a collage of four [paintings](#) by [William Blake](#).

Following the usual procedure for investigating images, it was noticed that this image contains a message, again, extractable by outguess. The following is its content, omitting the PGP header and signature:



```

e = 65537
n = 75579125746085351644267182920580212556413102071876330957950694457000592\
    10248050757270234679993673844203148013173091173786572116639

- -----BEGIN COMPRESSED RSA ENCRYPTED MESSAGE-----
Version: 1.99
Scheme: Crypt::RSA::ES::OAEP

eJwBswBM/zEwADE2MgBDeXBoZXJ0ZXh0LE2jxJS1EzMc80kOK+hralGKnXgQKQgVitiY8NgA7kxn
2u8jNQDvlu0uymNNiu6XVCn66axGH0IZ9w4Af3K/yRgjObsfA1Q7QqpXNALJ9FFPgYl5rh07cBP
M9kbSH6DynU/5cYgQod2KymjWcIvKx3FkjV4UOGakDnBfleQp1uwvn3KxDVwTyZPqbMnZvOA06Ec
AfKtyz1hEK/UBXkeMeVrnV5SQQ==
=yTUShDMKN65aPaKAR0OU8g==
- -----END COMPRESSED RSA ENCRYPTED MESSAGE-----

```

The message can be separated into two parts. The first part, spanning the first 3 rows, declares two values, an N and an e . The next part, hugged by "*BEGIN COMPRESSED RSA ENCRYPTED MESSAGE*", contains information about an encryption scheme, also called a chiffré, as well as data encoded in [base64](#). Base64 is a scheme to encode unprintable bytes into printable characters.

The *Scheme* line tells us that the following message is encrypted using the cipher RSA. The next step was clear: decrypt the message. To do that, we needed something we didn't have: the private key. A brief explanation is in order.

A Brief Overview of RSA

RSA is a moderately complex cipher to understand; [Numberphile](#) provides a good introduction to the topic. Its main advantage is that, as a [public key encryption](#) scheme, it allows sharing of encryption keys without transmitting the key in plaintext or agreeing on a key in advance. Public key cryptography solves the problem of how two parties can communicate securely without a pre-existing secure channel of communication.

In RSA, N and e are variables commonly used in the mathematical aspects of the cipher. In fact, together they constitute the public key of an RSA keypair. The public and private keys are mathematically related. Without going too much into detail, they both are related to N . N is the product of two large primes, called p and q . If an attacker can factor N , which is publicly available, into its two prime factors p and q , then he can calculate the corresponding private key.

In reality, factoring large integers that have only two prime factors is a [computationally hard problem](#). In RSA, an N with 2048 or 4096 bits is typically used as the large size provides enough complexity to make the factorization of p and q computationally infeasible in the short term.

Finding the Private Key (or, Brute Forcing RSA)

To our luck, the N used to encrypt the message was far smaller than 2048 or 4096 bits. The N we were given was 432 bits (130 decimal digits) long.

The solving community exhausted a lot of options in an attempt to find the private key. They searched for suspicious information in the data provided until that point, investigated images, brainstormed correlations, followed connections and so forth. As time went on and nothing was discovered, the solvers began to discuss the worst case scenario: finding p and q via brute computational force.

After a while it was agreed that a parallelized approach would render the best results, as a breakthrough in finding the hidden data would not impede on factorizing the number and vice versa. So a small group of people banded together to think about the most effective way to share their computing power in order to factor N . It was decided that a distributed approach would be the only feasible option, since even a 432 bit key requires an enormous computational effort not suited for a single processor. However, distributing the workload could achieve results in way less time, and the community was eager to help. We quickly agreed on [cado-nfs](#), a program designed to obtain the prime factors of large integers using a distributed network of machines (convenient to say the least). After about 8 hours of debugging, fixing, patching and testing as well as additional 9 hours of distributively working on the prime, the results were in:

```
p = 97513779050322159297664671238670850085661086043266591739338007321
q = 77506098606928780021829964781695212837195959082370473820509360759
```

These individuals successfully brute forced a 432-bit RSA key. Make no mistake -- this is an incredible feat for a group of people spread across the world with consumer-grade hardware. At this point, the writer would like to take a moment to express their gratitude for the people participating in this effort. Without the people donating their time, efforts and resources, this would have taken a lot longer than originally anticipated. Also, a special thanks needs to go to the people who managed the servers, especially the one unnamed person who rented a server from Amazon to complete the last, locally computed phase of the calculation. With all words of thanks being said, let's continue.

Putting The Private Key To Use

Now that we have the information, we have to make something out of it. To make sure we do it right, it would be wise to simply do what cicada did to encrypt the message and reverse it. We also have a clue for this in the outguessed message. The line

```
Scheme: Crypt::RSA::ES::OAEP
```

contains a dead giveaway. The programming language [Perl](#) uses [double colons](#) to achieve modularization. Therefore, we can safely conclude that a Perl program was used to encrypt this message. A decryption program in Perl using the found p and q values is available [here](#).

The Second Onion

This onion has since gone offline.

The Growing String

After the successful decryption of the RSA message, we had a single resource:

```
cu343l33nqaekrnw.onion
```

This is the address of a Tor Hidden Service. Upon visiting the service, the following document was displayed:

```
<!--Patience is a virtue-->
634292ba49fe336edada779a34054a335c2ec12c8bbaed4b92dcc05efe98f76abffdc2389bdb9de2c
f20c009acdc1945ab095a52609a5c219afd5f3b3edf10fcb25950666dfe8d8c433cd10c0b4c72efdf
e12c6270d5cfde291f9cf0d73cb1211140136e4057380c963d70c76948d9cf6775960cf98fbafa435
c44015c5959837a0f8d9f46e094f27c5797b7f8ab49bf28fa674d2ad2f726e197839956921dab2972
4cd48e1a81fc9bab3565f7513e3e368cd0327b47cf595afebb78d6b5bca92ba021cd6734f4362a0b3
41f359157173b53d49ea5dff5889d2c9de6b0d7e8c615286ce596bfa83f50b6eeabd153aaf50cd75f
39929ba11fb0f8e8d611442846
```

The page appeared to be static. However, after some time, we noticed that the string was slowly growing. Every few minutes, two characters were appended to the end of the string. This process continued for approximately 23 hours. The time intervals between new bytes were found to be multiples of five. Various users recorded the minutes between updates, the time that they occurred, and the data appended at those times.

<http://pastebin.com/5bTLHqCN>

<http://imgur.com/lTRRxTT>

<http://pastebin.com/qn8jmPJr> (GMT +1)

<http://i.imgur.com/prAeqPS.png>

The above datasets are most likely not complete, and no guarantee of accuracy is made.

After 23 hours, the process stopped and no more characters/bytes were appended to the string. The final string was:

```
634292ba49fe336edada779a34054a335c2ec12c8bbaed4b92dcc05efe98f76abffdc2389bdb9
de2cf20c009acdc1945ab095a52609a5c219afd5f3b3edf10fcb25950666dfe8d8c433cd10c0b
4c72efdfef12c6270d5cfde291f9cf0d73cb1211140136e4057380c963d70c76948d9cf6775960
cf98fbafa435c44015c5959837a0f8d9f46e094f27c5797b7f8ab49bf28fa674d2ad2f726e197
839956921dab29724cd48e1a81fc9bab3565f7513e3e368cd0327b47cf595afebb78d6b5bca92
ba021cd6734f4362a0b341f359157173b53d49ea5dff5889d2c9de6b0d7e8c615286ce596bfa8
3f50b6eeabd153aaf50cd75f39929ba11fb0f8e8d611442846
```

This string is 512 characters long. We assume it is hexadecimal.

However, about an hour after the string finished growing, at approximately 05:31:40 GMT the document changed. The old 512-character code was gone, along with the HTML comment. In its place was the following document:

<https://infotomb.com/oyfhl.txt>

backup: [sprunge](#)

We link to it because it is very large.

Note that the HTML comment was changed from

```
<!--Patience is a virtue-->
```

to

```
<!--761-->
```

This is significant. We noticed that, by applying the Gematrius Primus from the 2013 puzzle to the phrase 'Patience is a virtue' and summing the result, one obtains 761. The number is also a palindromic prime.

The new string was 3641299 (?) characters long. We noticed that it contained a significant amount of repeated text.

Analysis of this [new string](#) revealed that it contained three JPEG image files. This was discovered when we:

1. Converted the string into its binary representation
2. Flipped all the bits (i.e. 0 becomes 1, 1 becomes 0) [Note: this is equivalent to XOR 111111]

Looking at that XOR'd binary string, we noticed that the first two bytes were

```
0xFF 0xD8
```

Wikipedia tells us that these are the first two bytes of a JPEG image. We walked through the data left to right, and later on in the string, we discovered the same byte sequence again. This indicated the presence of more than one JPEG image. We analysed the rest of the string in a similar fashion, and discovered a total of three JPEG images. The third JPEG image was in reverse order, and so had to be un-reversed.

For further clarification, the order was this (.... = JPEG data, [] = one complete image):

```
[0xFF 0xD8.....][0xFF 0xD8.....][.....0xD8 0xFF]
```

After transforming the bytes into their proper JPEG format (note that you must reverse the byte sequence of the third image before doing this):

```
dd if=onioninvert.bin of=onion1.jpg bs=1 skip=0 count=168876
```

```
dd if=onioninvert.bin of=onion2.jpg bs=1 skip=168876 count=1476614
```

```
dd if=onioninvert.bin of=onion3rev.jpg  
bs=1 skip=1645490 count=175159
```

we obtained the following three images:

Liber Primus

Liber Primus:

Intus:

Chapter 1

Intus

Runes:



Image Analysis

Each of the images contain hidden messages and other information. These were found and analyzed simultaneously. As a result, this section does not proceed in chronological order.

Intus

```
outguess -r intus.jpg out.txt
```

yields <https://infotomb.com/esd78.txt>

Liber Primus

```
outguess -r liber_primus.jpg out.txt
```

yields <https://infotomb.com/hb0ba.txt>

Runes

```
outguess -r runes.jpg out.txt
```

yields <https://infotomb.com/vmtyf.txt>

Here we break to explain something:

Below are two images. The first image is from 2013's puzzle and was obtained through use of XOR. It was used to solve portions of that year's puzzle. The second was created by puzzle solvers in 2014 and is derived from the first. Here we refer to 2013's image, which was released by Cicada 3301, as 'Gematria Primus 2013'. We refer to 2014's image as 'Gematria Primus 2014'. **It is paramount to understand that 2013's image was created by Cicada 3301, while 2014's was created by puzzle solvers and was created from rearranging 2013's image.**

2013 gematria primus that created
by cicada3301:

Gematria Primus

an order and a value as revealed through 3301

Gematria Primus

Substitution key for decrypting Runes Warning 3301

F	F	↔	EA	T
h	U	↔	IA/O	X
f	TH	↔	Y	B
P	O	↔	AE	f
k	R	↔	A	f
L	C/K	↔	D	M
X	G	↔	OE	V
f	W	↔	NG/NG	X
H	H	↔	L	f
i	N	↔	M	M
I	I	↔	E	H
+	J	↔	B	f
z	EO	↔	T	f
K	P	↔	Z/S	h
T	X	↔	X	f
h	S/Z	↔	f	K
f	T	↔	EO	f
B	B	↔	J	f
H	E	↔	I	f
M	M	↔	N	f

'2014s was
created by puzzle
solvers and was

created from rearranging 2013's image

We now return to the puzzle. Using the Gematria Primus 2013, the runes in the 'Runes' image become:

```
R  NGRA  
  
JIHEIIAI MAEYW EAAAEN  
  
YEP JAEAED IXDISEO NGLREO THAEIA  
  
DMAENG EOAE JI EOIAI EOIPEO YI D  
  
MAENGHICOEI EAEMC THAEIAA EOAIAY IX  
  
SIAEIMDI THAEIAA CFY CAE MAEEO ICEEO AE  
  
A  DLRWI YEP JAEAED AEA YI NICCROEI  
  
DAEMEOREMIC NGEYEM IEYIA YI NGA  
  
ACC AEA YIEA MIANJIAC EAAEA RHH E  
  
C  CRDAIC
```

This stumped us for a short while. Then, someone created Gematria Primus 2014 by rearranging Gematria Primus 2013. From Gematria Primus 13, take the left block of three columns and stack it on top of the right block of three columns. Then, find the letter you wish to decode. Find the position of that letter in the vertical list, beginning at the top. Take that number and find the character that many characters into the list, from the bottom up. Applying this process to each letter/rune yields (newlines added for readability):

```
A  WARNING  
BELIEVE NOTHING FROM THIS BOOK  
EXCEPT WHAT YOU KNOW TO BE TRUE
```

TEST THE KNOWLEDGE
FIND YOUR TRUTH
EXPERIENCE YOUR DEATH
DO NOT EDIT OR CHANGE THIS BOOK
OR THE MESSAGE CONTAINED WITHIN
EITHER THE WORDS OR THEIR NUMBERS
FOR ALL IS SACRED

By substituting each character for its respective value in Gematria Primus 2013, we noticed that the sum of each line adds up to a prime (emirps marked by *):

A WARNING

BELIEVE NOTHING FROM THIS BOOK	= 757*
EXCEPT WHAT YOU KNOW TO BE TRUE	= 1009*
TEST THE KNOWLEDGE	= 691
FIND YOUR TRUTH	= 353*
EXPERIENCE YOUR DEATH	= 769*
DO NOT EDIT OR CHANGE THIS BOOK	= 911*
OR THE MESSAGE CONTAINED WITHIN	= 1051*
EITHER THE WORDS OR THE NUMBERS	= 859
FOR ALL IS SACRED	= 677

The 5 Gram Message

Shortly after the warning in the previous section was found, it was discovered that by XORing the hexadecimal messages hidden within each of the three images the following message resulted:

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

IDGTK UML00 ARWOE RTHIS UTETL HUTIA TSLLO
UIMNI TELNJ 7TFYV OIUUAU SNOCO 5JI4M EODZZ

Good luck.

3301

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

iQIcBAEBAGAGBQJSy23PAAoJEBgfAeV6NQkPeJwP/0IoafJlSbmhD+KNbL5I2EdH
jgPRnZNrKCymPWFSIwlqs6ujuw6VnW/rfnOD+df4kpzoAwEFfZDcRnBVsvIzOJ31
Txj9jXD22ki/CNRY88NyIzW9fjKs+iOylsa7Tx+6PBb3ndoYNEwnQwLIq3K4S3kQ
tgMzE3LiVq2pQwqfNdN+zGqcq7POEs0GmnL1aNpqU+Wrba4gSfoWwQBWUDv3S/s8
vY0hEqhWNd76wphig6hH6OyIaX/tleYfcsSYhzAE5oKKahGr1E7cX1GBpHCiRlWM
ZwNaGVArQAkyEzT++tmF01O9h218CiTUFoBM/Zxyra7vxI2UOYS/pLonuV+eXARY
YfPhAZZxfk3bUWXcxioRukFSY2+xNdPfuBIT8rcJqa1kPJ0zeZVC/IcwhA2mmG41
3ltiVcDnQrZgz6Im3/ugFg8bqW12qqZ6XizRP3EXm4EnyhpFKZnXKPLEOvPKCj6j
1kYCrLmGtTTPFx79fZfryGXQIEAmipRbjVS5sVbUCfgmqUagmdU6v9VI53n6+r0J
b2amxREA+2MflkEoVJUaLQJ1rKZLFFJ9J17zUaXKMl1sDBWXJS4Mb54o2+8bkEcM
3cP+16XV9pf2wZBkJE0AwoXI4L8JEYjNZZcGSLy8BojlAupX3Fg9KKt71XXrm9FD
tuBhMYWo/TDz+4UzLB+I

```
=57tj
-----END PGP SIGNATURE-----
```

After some trial and error analysis, it was discovered that this ciphertext had been created using a simple column transposition cipher. By arranging the ciphertext into 14 columns like so:

0	1	2	3	4	5	6	7	8	9	10	11	12	13
I	D	G	T	K	U	M	L	O	O	A	R	W	O
E	R	T	H	I	S	U	T	E	T	L	H	U	T
I	A	T	S	L	L	O	U	I	M	N	I	T	E
L	N	J	7	T	F	Y	V	O	I	U	A	U	S
N	O	C	O	5	J	I	4	M	E	O	D	Z	Z

And reordering the columns like so:

2	8	9	1	12	13	11	4	5	7	3	0	6	10
G	O	O	D	W	O	R	K	U	L	T	I	M	A
T	E	T	R	U	T	H	I	S	T	H	E	U	L
T	I	M	A	T	E	I	L	L	U	S	I	O	N
J	O	I	N	U	S	A	T	F	V	7	L	Y	U
C	M	E	O	Z	Z	D	5	J	4	O	N	I	O

A message is obtained:

```
GOOD WORK
ULTIMATE TRUTH IS THE ULTIMATE ILLUSION
JOIN US AT FV7LYUCMEOZZD5J4ONIO
```

Assuming that the final N was omitted in order to fit the bounds of the column transposition cipher, and applying proper formatting, the following Tor hidden service address is obtained:

```
fv7lyucmeozzd5j4.onion
```

And off we went.

The Third Onion

This onion has since been taken offline. In this section the puzzle fractures into several directions. The author has chosen not to subdivide this section and instead opts for pure chronological formatting. Events in this section are detailed in the exact order they occurred.

The first visitor to this hidden service was greeted with a blank page. The page remained blank for a short time and then changed to the following document:

```
<!--1033-->
87de5b7fa2
```

As with the RSA onion, the string slowly grew over time, with two new characters (one byte) being added at widely varying intervals. Some timing data was collected for further analysis.

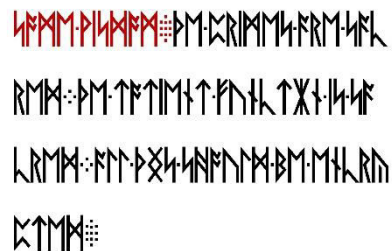
In the meantime, a solver ran the tool [DirBuster](#) against the hidden service and discovered an apparent misconfiguration of the backend Apache server. The server was leaking a system status page. Whether this was intentional is unknown. However, shortly after the page was discovered, it appears that Cicada was alerted that the status page had been discovered because the content of that page changed. Appended to the end of the server status was yet another very long string. This string was found to contain two image files in a similar ordering as the RSA onion, except that there was some data between them (OOB or Out Of Bounds data):

```
[0xFF 0xD8.....] [Data in between JPGs] [.....0xD8 0xFF]
```

After building the first JPG from the hex:

```
xxd -p -r < server-status.hex > server-status.jpg
```

One obtains the image shown below on the left. Doing the same for the reversed copy of the second JPG yields the same image as the first, shown -- except for that OOB data.



272	138	WMMW	131	151
MMRYY	MMRYY	MM	MM	18
226	MMW	MM	245	MMW
18	MMW	MM	MMW	MMRYY
151	131	MM	138	272

Comparing the first and second images

```
cmp -l server-status.jpg rev.server-status.jpg
```

one obtains the OOB data:

```
a02373230202020202833313020202020213433302020202021333130202020202135313a06363
330202020202939313020202020203331302020202020323330202020202028313a06323230202
020202534323020202020202139302020202025343230202020202632323a08313020202020203
2333020202020203331302020202029393130202020202636333a0135313020202020213331302
```

```
02020202134333020202020283331302020202022373230a0a
```

Note that all of these bytes are within the printable range of ASCII characters, and many of them appear to be ASCII for digits (e.g. 0x30, 0x39).

Converting this string to binary:

```
xxd -b oob.hex oob.bin
```

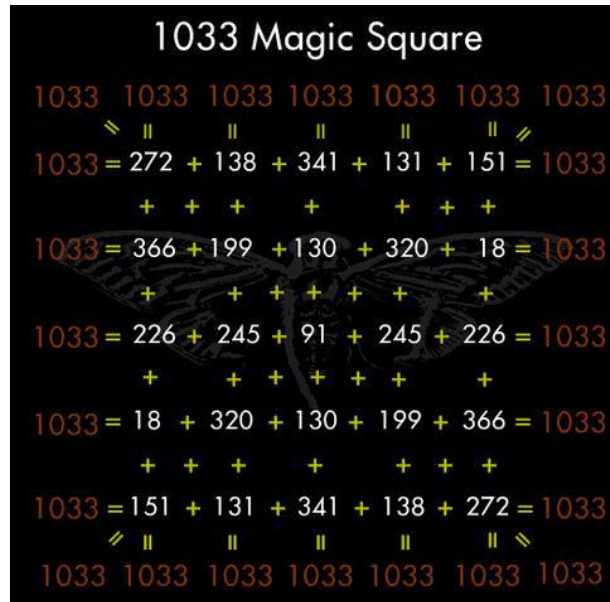
and reversing that::

```
xxd -r oob.bin oob-rev.bin
```

we obtain:

272	138	341	131	151
366	199	130	320	18
226	245	91	245	226
18	320	130	199	366
151	131	341	138	272

Among other things, this is a magic square whose magic number is -- you guessed it -- 1033.



For those following along, here's a nice one-liner to get that matrix from the original image:

```
dd if=server-status.jpg bs=1 skip=$((0x00521e4)) count=357 status=noxfer | rev |  
xxd -p -r
```

Runes were unencrypted. With matrix and Gematria Primus 2013 in hand, we began to interpret the runes in the image. The runes in the upper half of the image read:

SOME WISDOM
THE PRIMES ARE SACRED
THE TOTIENT FUNCTION IS SACRED
ALL THINGS SHOULD BE ENCRYPTED

In the bottom half of the image, the red runes read:

KNOW THIS:

The table underneath, translated to value form using Gematria Primus 2013, yields:

272	138	shadows	131	151
aethereal	buffers	void	carnal	18
226	obscura	form	245	mobius
18	analog	void	mournful	aethereal
151	131	cabal	138	272
272	138	341	131	151
366	199	130	320	18
226	245	91	245	226
18	320	130	199	366
151	131	341	138	272

which is the exact same matrix found earlier from the OOB data.

About a day went by with little activity other than speculation on the meaning of this matrix. Then the string from the main page stopped growing. According to the HTTP header, the final update occurred on January 11 at 01:09:01 GMT. The final document was:

```
<!--1033-->
87de5b7fa26ab85d2256c453e7f5bc3ac7f25ee743297817febd7741ededf07ca0c7e8b1788ea4131
441a8f71c63943d8b56aea6a45159e2f59f9a194af23eabf9de0f3123c041c882d5b7e03e17ac49b
e67cef29fbc7786e3bda321a176498835f6198ef22e81c30d44281cd217f7a46f58c84dd7b29b9414
03ecd75c0c735d20266121f875aa8dec28f32fc153b1393e143fc71616945eea3c10d6820bd631cf7
75cf3c1f27925b4a2da655f783f7616f3359b23cff6fb5cb69bcb745c55dff439f7eb6a4094bd302b
65a84360a62f94c8b010250fcc431c190d6ed8cc8a3bfce37dddb24b93f502ad83c5fa21923189d8b
e7a6127c4105fcf0e5275286f2
```

The final string was 512 characters (assuming hex, 256 bytes or equivalently 2048 bits). It was observed that this string matched the growing string from the previous onion.

On January 11 at 10:07 UTC, the Apache server status page changed once again to display a new hexadecimal string:

<https://infotomb.com/lags9.txt>

This string contained the following two images in the same style as before

[0xFF 0xD8.....] [.....0xD8 0xFF]



First Image

Outguessing the first image yields a signed message:

<https://infotomb.com/t5uuz.txt>

The hex string from that message encodes a JPG image:

ÞM̥J̥M̥R5N̥X̥R̥X̥3Y̥M̥J̥X̥

The runes in this image were solved later (read on).

Translating the runes from the first image with Gematria Primus yields :

```
uWGsSfc rSugpWW fwxtclW ym WS tcnF
GmXXmmw FpdGXr oW Xmi ff euG SuF yp
rF ipF cF Fnw bxmd rXi fpc SSFc rTp
fjmo ScwX bFw bWls ry jF r mcTSFtcpw
mgS cGpu Sc rew Xpi bybx flir rm cgb
yr cfu TpXjwtfW jgb FTffWpT pfax
jmepGsosm wSjl wxuT FwmT dyjc
sXxoGrmbw rmi dyjc xhuypL jGymfjpSuX
wSj dwxu alasmXSx Fm Xmi py Fmuf
```

After some time with trial and error, it was discovered that this ciphertext had been created with a Vignere cipher.

The key was DIUINITY (divinity) and used the reversed gematria as shown here

Gematria Primus

Substitution key for decrypting Runes Warning 3301

Þ	Ɔ	↔	Yð	𐀀
𐀁	U	↔	OU/YI	𐀂
Ɔ	TH	↔	A	𐀃
Þ	O	↔	ZV	𐀄
𐀅	E	↔	V	𐀅
L	C/K	↔	O	𐀆
Y	O	↔	ZO	𐀇
Þ	W	↔	ONU/DN	𐀈
𐀉	H	↔	I	𐀉
𐀊	N	↔	W	𐀊
𐀋	I	↔	I	𐀋
𐀌	J	↔	U	𐀌
𐀍	EO	↔	L	𐀍
𐀎	P	↔	Z/S	𐀎
T	X	↔	X	𐀏
𐀐	S/Z	↔	𐀐	𐀐
𐀑	T	↔	OO	𐀑
𐀒	B	↔	S	𐀒
𐀓	E	↔	I	𐀓
𐀔	M	↔	N	𐀔
𐀕	L	↔	W	𐀕
𐀖	NG/VNG	↔	M	𐀖
𐀗	OE	↔	O	𐀗
𐀘	D	↔	C/K	𐀘
Þ	A	↔	𐀙	𐀙
𐀚	AE	↔	O	𐀚
𐀛	F	↔	H	𐀛
𐀜	IA/IO	↔	U	𐀜
T	EA	↔	𐀝	𐀝

Every time you reached an F it was left unencrypted, and it reset the running key.

Reversing the cipher yields:

```
WELCOME:
WELCOME, PILGRIM TO THE GREAT JOURNEY
TOWARD THE END OF ALL THINGS.
IT IS NOT AN EASY TRIP, BUT FOR THOSE WHO
FIND THEIR WAY HERE IT IS A NECESSARY ONE.
ALONG THE WAY YOU WILL FIND AN END TO ALL
STRUGGLE AND SUFFERING, YOUR INNOCENCE, YOUR
ILLUSIONS, YOUR CERTAINTY, AND YOUR REALITY.
ULTIMATELY, YOU WILL DISCOVER AN END TO SELF.
```

Second Image

Outguessing the second image yields garbage output.

The runes on it translate to:

```
my yS Fxrjse ewn djusxytetm Sry ds neFdX pbunWGjXF jgb pTx pnwwilmF lpbuoWX rXWf
rrSjm rmi dyj hlFu juXlTW SjoSrrm umsc WS liFFcl wi lt peup WXpTtb tme ulole Sjp
uW lcg WgsXtm
bmrTfp wrj rxc G jWQ je ym dyjcFXuf pfa ccW r ujr ambp gpbunWGf nxe ygiWGumtcgWW
jF bpwd fyX Fuf Sjp xlTWa lT cyuX
ce lFSixTsFhF Tyflcer pfax rbe Fcbf
```

After reversing the Vignere cipher:

```
IT IS THROUGH THIS PILGRIMAGE THAT WE SHAPE
OURSELVES AND OUR REALITIES.
JOURNEY DEEP WITHIN AND YOU WILL ARRIVE OUTSIDE.
LIKE THE INSTAR, IT IS ONLY THROUGH GOING
WITHIN THAT WE MAY EMERGE:

WISDOM:
YOU ARE A BEING UNTO YOURSELF.
YOU ARE A LAW UNTO YOURSELF.
EACH INTELLIGENCE IS HOLY.
FOR ALL THAT LIVES IS HOLY.
```

The red footer of that page reads

```
:AN INSTRUCTION: COMMAND YOUR OWN SELF :
```

Back to the server page:

Fourteen minutes after the status page update containing the two above images, on January 11 at 10:22 UTC the status page changed once again:

<https://infotomb.com/hw015.txt>

The first image from the previous update remained intact in the new string. However, the data of the second was replaced almost entirely with different hexademical. Attempting to render it as a JPG yields a corrupt and incomplete image.

Outguessed Image Solved

The runes in the small outguessed image obtained from the first new page were also encoded with a Vignere cipher. Using the key

```
welcome pilgrim to the
```

we obtained the Vignere offsets

```
22, 11, 9, 24, 26, 10, 11, 16, 19, 9, 23, 25, 19, 10, 13, 26, 27, 11
```

and using these offsets to decode the original runetext we obtained

```
A U O W Y F X L 5 L C S F J 3 N O N I A N
```

which formatted as a hidden service address yields

```
avowyfgl5lkz fj3n.onion
```

avowyfgl5lkz fj3n.onion
𐀀𐀁𐀂𐀃𐀄𐀅𐀆𐀇𐀈𐀉𐀊𐀋𐀌𐀍𐀎𐀏𐀐𐀑𐀒𐀓𐀔𐀕𐀖𐀗𐀘𐀙𐀚𐀛𐀜𐀝𐀞𐀟𐀠𐀡𐀢𐀣𐀤𐀥𐀦𐀧𐀨𐀩𐀪𐀫𐀬𐀭𐀮𐀯𐀰𐀱𐀲𐀳𐀴𐀵𐀶𐀷𐀸𐀹𐀺𐀻𐀼𐀽𐀾𐀿𐁀𐁁𐁂𐁃𐁄𐁅𐁆𐁇𐁈𐁉𐁊𐁋𐁌𐁍𐁎𐁏𐁐𐁑𐁒𐁓𐁔𐁕𐁖𐁗𐁘𐁙𐁚𐁛𐁜𐁝𐁞𐁟𐁠𐁡𐁢𐁣𐁤𐁥𐁦𐁧𐁨𐁩𐁪𐁫𐁬𐁭𐁮𐁯𐁰𐁱𐁲𐁳𐁴𐁵𐁶𐁷𐁸𐁹𐁺𐁻𐁼𐁽𐁾𐁿𐂀𐂁𐂂𐂃𐂄𐂅𐂆𐂇𐂈𐂉𐂊𐂋𐂌𐂍𐂎𐂏𐂐𐂑𐂒𐂓𐂔𐂕𐂖𐂗𐂘𐂙𐂚𐂛𐂜𐂝𐂞𐂟𐂠𐂡𐂢𐂣𐂤𐂥𐂦𐂧𐂨𐂩𐂪𐂫𐂬𐂭𐂮𐂯𐂰𐂱𐂲𐂳𐂴𐂵𐂶𐂷𐂸𐂹𐂺𐂻𐂼𐂽𐂾𐂿𐃀𐃁𐃂𐃃𐃄𐃅𐃆𐃇𐃈𐃉𐃊𐃋𐃌𐃍𐃎𐃏𐃐𐃑𐃒𐃓𐃔𐃕𐃖𐃗𐃘𐃙𐃚𐃛𐃜𐃝𐃞𐃟𐃠𐃡𐃢𐃣𐃤𐃥𐃦𐃧𐃨𐃩𐃪𐃫𐃬𐃭𐃮𐃯𐃰𐃱𐃲𐃳𐃴𐃵𐃶𐃷𐃸𐃹𐃺𐃻𐃼𐃽𐃾𐃿𐄀𐄁𐄂𐄃𐄄𐄅𐄆𐄇𐄈𐄉𐄊𐄋𐄌𐄍𐄎𐄏𐄐𐄑𐄒𐄓𐄔𐄕𐄖𐄗𐄘𐄙𐄚𐄛𐄜𐄝𐄞𐄟𐄠𐄡𐄢𐄣𐄤𐄥𐄦𐄧𐄨𐄩𐄪𐄫𐄬𐄭𐄮𐄯𐄰𐄱𐄲𐄳𐄴𐄵𐄶𐄷𐄸𐄹𐄺𐄻𐄼𐄽𐄾𐄿𐅀𐅁𐅂𐅃𐅄𐅅𐅆𐅇𐅈𐅉𐅊𐅋𐅌𐅍𐅎𐅏𐅐𐅑𐅒𐅓𐅔𐅕𐅖𐅗𐅘𐅙𐅚𐅛𐅜𐅝𐅞𐅟𐅠𐅡𐅢𐅣𐅤𐅥𐅦𐅧𐅨𐅩𐅪𐅫𐅬𐅭𐅮𐅯𐅰𐅱𐅲𐅳𐅴𐅵𐅶𐅷𐅸𐅹𐅺𐅻𐅼𐅽𐅾𐅿𐆀𐆁𐆂𐆃𐆄𐆅𐆆𐆇𐆈𐆉𐆊𐆋𐆌𐆍𐆎𐆏𐆐𐆑𐆒𐆓𐆔𐆕𐆖𐆗𐆘𐆙𐆚𐆛𐆜𐆝𐆞𐆟𐆠𐆡𐆢𐆣𐆤𐆥𐆦𐆧𐆨𐆩𐆪𐆫𐆬𐆭𐆮𐆯𐆰𐆱𐆲𐆳𐆴𐆵𐆶𐆷𐆸𐆹𐆺𐆻𐆼𐆽𐆾𐆿𐇀𐇁𐇂𐇃𐇄𐇅𐇆𐇇𐇈𐇉𐇊𐇋𐇌𐇍𐇎𐇏𐇐𐇑𐇒𐇓𐇔𐇕𐇖𐇗𐇘𐇙𐇚𐇛𐇜𐇝𐇞𐇟𐇠𐇡𐇢𐇣𐇤𐇥𐇦𐇧𐇨𐇩𐇪𐇫𐇬𐇭𐇮𐇯𐇰𐇱𐇲𐇳𐇴𐇵𐇶𐇷𐇸𐇹𐇺𐇻𐇼𐇽𐇾𐇿𐈀𐈁𐈂𐈃𐈄𐈅𐈆𐈇𐈈𐈉𐈊𐈋𐈌𐈍𐈎𐈏𐈐𐈑𐈒𐈓𐈔𐈕𐈖𐈗𐈘𐈙𐈚𐈛𐈜𐈝𐈞𐈟𐈠𐈡𐈢𐈣𐈤𐈥𐈦𐈧𐈨𐈩𐈪𐈫𐈬𐈭𐈮𐈯𐈰𐈱𐈲𐈳𐈴𐈵𐈶𐈷𐈸𐈹𐈺𐈻𐈼𐈽𐈾𐈿𐉀𐉁𐉂𐉃𐉄𐉅𐉆𐉇𐉈𐉉𐉊𐉋𐉌𐉍𐉎𐉏𐉐𐉑𐉒𐉓𐉔𐉕𐉖𐉗𐉘𐉙𐉚𐉛𐉜𐉝𐉞𐉟𐉠𐉡𐉢𐉣𐉤𐉥𐉦𐉧𐉨𐉩𐉪𐉫𐉬𐉭𐉮𐉯𐉰𐉱𐉲𐉳𐉴𐉵𐉶𐉷𐉸𐉹𐉺𐉻𐉼𐉽𐉾𐉿𐊀𐊁𐊂𐊃𐊄𐊅𐊆𐊇𐊈𐊉𐊊𐊋𐊌𐊍𐊎𐊏𐊐𐊑𐊒𐊓𐊔𐊕𐊖𐊗𐊘𐊙𐊚𐊛𐊜𐊝𐊞𐊟𐊠𐊡𐊢𐊣𐊤𐊥𐊦𐊧𐊨𐊩𐊪𐊫𐊬𐊭𐊮𐊯𐊰𐊱𐊲𐊳𐊴𐊵𐊶𐊷𐊸𐊹𐊺𐊻𐊼𐊽𐊾𐊿𐋀𐋁𐋂𐋃𐋄𐋅𐋆𐋇𐋈𐋉𐋊𐋋𐋌𐋍𐋎𐋏𐋐𐋑𐋒𐋓𐋔𐋕𐋖𐋗𐋘𐋙𐋚𐋛𐋜𐋝𐋞𐋟𐋠𐋡𐋢𐋣𐋤𐋥𐋦𐋧𐋨𐋩𐋪𐋫𐋬𐋭𐋮𐋯𐋰𐋱𐋲𐋳𐋴𐋵𐋶𐋷𐋸𐋹𐋺𐋻𐋼𐋽𐋾𐋿𐌀𐌁𐌂𐌃𐌄𐌅𐌆𐌇𐌈𐌉𐌊𐌋𐌌𐌍𐌎𐌏𐌐𐌑𐌒𐌓𐌔𐌕𐌖𐌗𐌘𐌙𐌚𐌛𐌜𐌝𐌞𐌟𐌠𐌡𐌢𐌣𐌤𐌥𐌦𐌧𐌨𐌩𐌪𐌫𐌬𐌭𐌮𐌯𐌰𐌱𐌲𐌳𐌴𐌵𐌶𐌷𐌸𐌹𐌺𐌻𐌼𐌽𐌾𐌿𐍀𐍁𐍂𐍃𐍄𐍅𐍆𐍇𐍈𐍉𐍊𐍋𐍌𐍍𐍎𐍏𐍐𐍑𐍒𐍓𐍔𐍕𐍖𐍗𐍘𐍙𐍚𐍛𐍜𐍝𐍞𐍟𐍠𐍡𐍢𐍣𐍤𐍥𐍦𐍧𐍨𐍩𐍪𐍫𐍬𐍭𐍮𐍯𐍰𐍱𐍲𐍳𐍴𐍵𐍶𐍷𐍸𐍹𐍺𐍻𐍼𐍽𐍾𐍿𐎀𐎁𐎂𐎃𐎄𐎅𐎆𐎇𐎈𐎉𐎊𐎋𐎌𐎍𐎎𐎏𐎐𐎑𐎒𐎓𐎔𐎕𐎖𐎗𐎘𐎙𐎚𐎛𐎜𐎝𐎞𐎟𐎠𐎡𐎢𐎣𐎤𐎥𐎦𐎧𐎨𐎩𐎪𐎫𐎬𐎭𐎮𐎯𐎰𐎱𐎲𐎳𐎴𐎵𐎶𐎷𐎸𐎹𐎺𐎻𐎼𐎽𐎾𐎿𐏀𐏁𐏂𐏃𐏄𐏅𐏆𐏇𐏈𐏉𐏊𐏋𐏌𐏍𐏎𐏏𐏐𐏑𐏒𐏓𐏔𐏕𐏖𐏗𐏘𐏙𐏚𐏛𐏜𐏝𐏞𐏟𐏠𐏡𐏢𐏣𐏤𐏥𐏦𐏧𐏨𐏩𐏪𐏫𐏬𐏭𐏮𐏯𐏰𐏱𐏲𐏳𐏴𐏵𐏶𐏷𐏸𐏹𐏺𐏻𐏼𐏽𐏾𐏿𐐀𐐁𐐂𐐃𐐄𐐅𐐆𐐇𐐈𐐉𐐊𐐋𐐌𐐍𐐎𐐏𐐐𐐑𐐒𐐓𐐔𐐕𐐖𐐗𐐘𐐙𐐚𐐛𐐜𐐝𐐞𐐟𐐠𐐡𐐢𐐣𐐤𐐥𐐦𐐧𐐨𐐩𐐪𐐫𐐬𐐭𐐮𐐯𐐰𐐱𐐲𐐳𐐴𐐵𐐶𐐷𐐸𐐹𐐺𐐻𐐼𐐽𐐾𐐿𐑀𐑁𐑂𐑃𐑄𐑅𐑆𐑇𐑈𐑉𐑊𐑋𐑌𐑍𐑎𐑏𐑐𐑑𐑒𐑓𐑔𐑕𐑖𐑗𐑘𐑙𐑚𐑛𐑜𐑝𐑞𐑟𐑠𐑡𐑢𐑣𐑤𐑥𐑦𐑧𐑨𐑩𐑪𐑫𐑬𐑭𐑮𐑯𐑰𐑱𐑲𐑳𐑴𐑵𐑶𐑷𐑸𐑹𐑺𐑻𐑼𐑽𐑾𐑿𐒀𐒁𐒂𐒃𐒄𐒅𐒆𐒇𐒈𐒉𐒊𐒋𐒌𐒍𐒎𐒏𐒐𐒑𐒒𐒓𐒔𐒕𐒖𐒗𐒘𐒙𐒚𐒛𐒜𐒝𐒞𐒟𐒠𐒡𐒢𐒣𐒤𐒥𐒦𐒧𐒨𐒩𐒪𐒫𐒬𐒭𐒮𐒯𐒰𐒱𐒲𐒳𐒴𐒵𐒶𐒷𐒸𐒹𐒺𐒻𐒼𐒽𐒾𐒿𐓀𐓁𐓂𐓃𐓄𐓅𐓆𐓇𐓈𐓉𐓊𐓋𐓌𐓍𐓎𐓏𐓐𐓑𐓒𐓓𐓔𐓕𐓖𐓗𐓘𐓙𐓚𐓛𐓜𐓝𐓞𐓟𐓠𐓡𐓢𐓣𐓤𐓥𐓦𐓧𐓨𐓩𐓪𐓫𐓬𐓭𐓮𐓯𐓰𐓱𐓲𐓳𐓴𐓵𐓶𐓷𐓸𐓹𐓺𐓻𐓼𐓽𐓾𐓿𐔀𐔁𐔂𐔃𐔄𐔅𐔆𐔇𐔈𐔉𐔊𐔋𐔌𐔍𐔎𐔏𐔐𐔑𐔒𐔓𐔔𐔕𐔖𐔗𐔘𐔙𐔚𐔛𐔜𐔝𐔞𐔟𐔠𐔡𐔢𐔣𐔤𐔥𐔦𐔧𐔨𐔩𐔪𐔫𐔬𐔭𐔮𐔯𐔰𐔱𐔲𐔳𐔴𐔵𐔶𐔷𐔸𐔹𐔺𐔻𐔼𐔽𐔾𐔿𐕀𐕁𐕂𐕃𐕄𐕅𐕆𐕇𐕈𐕉𐕊𐕋𐕌𐕍𐕎𐕏𐕐𐕑𐕒𐕓𐕔𐕕𐕖𐕗𐕘𐕙𐕚𐕛𐕜𐕝𐕞𐕟𐕠𐕡𐕢𐕣𐕤𐕥𐕦𐕧𐕨𐕩𐕪𐕫𐕬𐕭𐕮𐕯𐕰𐕱𐕲𐕳𐕴𐕵𐕶𐕷𐕸𐕹𐕺𐕻𐕼𐕽𐕾𐕿𐖀𐖁𐖂𐖃𐖄𐖅𐖆𐖇𐖈𐖉𐖊𐖋𐖌𐖍𐖎𐖏𐖐𐖑𐖒𐖓𐖔𐖕𐖖𐖗𐖘𐖙𐖚𐖛𐖜𐖝𐖞𐖟𐖠𐖡𐖢𐖣𐖤𐖥𐖦𐖧𐖨𐖩𐖪𐖫𐖬𐖭𐖮𐖯𐖰𐖱𐖲𐖳𐖴𐖵𐖶𐖷𐖸𐖹𐖺𐖻𐖼𐖽𐖾𐖿𐗀𐗁𐗂𐗃𐗄𐗅𐗆𐗇𐗈𐗉𐗊𐗋𐗌𐗍𐗎𐗏𐗐𐗑𐗒𐗓𐗔𐗕𐗖𐗗𐗘𐗙𐗚𐗛𐗜𐗝𐗞𐗟𐗠𐗡𐗢𐗣𐗤𐗥𐗦𐗧𐗨𐗩𐗪𐗫𐗬𐗭𐗮𐗯𐗰𐗱𐗲𐗳𐗴𐗵𐗶𐗷𐗸𐗹𐗺𐗻𐗼𐗽𐗾𐗿𐘀𐘁𐘂𐘃𐘄𐘅𐘆𐘇𐘈𐘉𐘊𐘋𐘌𐘍𐘎𐘏𐘐𐘑𐘒𐘓𐘔𐘕𐘖𐘗𐘘𐘙𐘚𐘛𐘜𐘝𐘞𐘟𐘠𐘡𐘢𐘣𐘤𐘥𐘦𐘧𐘨𐘩𐘪𐘫𐘬𐘭𐘮𐘯𐘰𐘱𐘲𐘳𐘴𐘵𐘶𐘷𐘸𐘹𐘺𐘻𐘼𐘽𐘾𐘿𐙀𐙁𐙂𐙃𐙄𐙅𐙆𐙇𐙈𐙉𐙊𐙋𐙌𐙍𐙎𐙏𐙐𐙑𐙒𐙓𐙔𐙕𐙖𐙗𐙘𐙙𐙚𐙛𐙜𐙝𐙞𐙟𐙠𐙡𐙢𐙣𐙤𐙥𐙦𐙧𐙨𐙩𐙪𐙫𐙬𐙭𐙮𐙯𐙰𐙱𐙲𐙳𐙴𐙵𐙶𐙷𐙸𐙹𐙺𐙻𐙼𐙽𐙾𐙿𐚀𐚁𐚂𐚃𐚄𐚅𐚆𐚇𐚈𐚉𐚊𐚋𐚌𐚍𐚎𐚏𐚐𐚑𐚒𐚓𐚔𐚕𐚖𐚗𐚘𐚙𐚚𐚛𐚜𐚝𐚞𐚟𐚠𐚡𐚢𐚣𐚤𐚥𐚦𐚧𐚨𐚩𐚪𐚫𐚬𐚭𐚮𐚯𐚰𐚱𐚲𐚳𐚴𐚵𐚶𐚷𐚸𐚹𐚺𐚻𐚼𐚽𐚾𐚿𐛀𐛁𐛂𐛃𐛄𐛅𐛆𐛇𐛈𐛉𐛊𐛋𐛌𐛍𐛎𐛏𐛐𐛑𐛒𐛓𐛔𐛕𐛖𐛗𐛘𐛙𐛚𐛛𐛜𐛝𐛞𐛟𐛠𐛡𐛢𐛣𐛤𐛥𐛦𐛧𐛨𐛩𐛪𐛫𐛬𐛭𐛮𐛯𐛰𐛱𐛲𐛳𐛴𐛵𐛶𐛷𐛸𐛹𐛺𐛻𐛼𐛽𐛾𐛿𐜀𐜁𐜂𐜃𐜄𐜅𐜆𐜇𐜈𐜉𐜊𐜋𐜌𐜍𐜎𐜏𐜐𐜑𐜒𐜓𐜔𐜕𐜖𐜗𐜘𐜙𐜚𐜛𐜜𐜝𐜞𐜟𐜠𐜡𐜢𐜣𐜤𐜥𐜦𐜧𐜨𐜩𐜪𐜫𐜬𐜭𐜮𐜯𐜰𐜱𐜲𐜳𐜴𐜵𐜶𐜷𐜸𐜹𐜺𐜻𐜼𐜽𐜾𐜿𐝀𐝁𐝂𐝃𐝄𐝅𐝆𐝇𐝈𐝉𐝊𐝋𐝌𐝍𐝎𐝏𐝐𐝑𐝒𐝓𐝔𐝕𐝖𐝗𐝘𐝙𐝚𐝛𐝜𐝝𐝞𐝟𐝠𐝡𐝢𐝣𐝤𐝥𐝦𐝧𐝨𐝩𐝪𐝫𐝬𐝭𐝮𐝯𐝰𐝱𐝲𐝳𐝴𐝵𐝶𐝷𐝸𐝹𐝺𐝻𐝼𐝽𐝾𐝿𐞀𐞁𐞂𐞃𐞄𐞅𐞆𐞇𐞈𐞉𐞊𐞋𐞌𐞍𐞎𐞏𐞐𐞑𐞒𐞓𐞔𐞕𐞖𐞗𐞘𐞙𐞚𐞛𐞜𐞝𐞞𐞟𐞠𐞡𐞢𐞣𐞤𐞥𐞦𐞧𐞨𐞩𐞪𐞫𐞬𐞭𐞮𐞯𐞰𐞱𐞲𐞳𐞴𐞵𐞶𐞷𐞸𐞹𐞺𐞻𐞼𐞽𐞾𐞿𐟀𐟁𐟂𐟃𐟄𐟅𐟆𐟇𐟈𐟉𐟊𐟋𐟌𐟍𐟎𐟏𐟐𐟑𐟒𐟓𐟔𐟕𐟖𐟗𐟘𐟙𐟚𐟛𐟜𐟝𐟞𐟟𐟠𐟡𐟢𐟣𐟤𐟥𐟦𐟧𐟨𐟩𐟪𐟫𐟬𐟭𐟮𐟯𐟰𐟱𐟲𐟳𐟴𐟵𐟶𐟷𐟸𐟹𐟺𐟻𐟼𐟽𐟾𐟿𐠀𐠁𐠂𐠃𐠄𐠅𐠆𐠇𐠈𐠉𐠊𐠋𐠌𐠍𐠎𐠏𐠐𐠑𐠒𐠓𐠔𐠕𐠖𐠗𐠘𐠙𐠚𐠛𐠜𐠝𐠞𐠟𐠠𐠡𐠢𐠣𐠤𐠥𐠦𐠧𐠨𐠩𐠪𐠫𐠬𐠭𐠮𐠯𐠰𐠱𐠲𐠳𐠴𐠵𐠶𐠷𐠸𐠹𐠺𐠻𐠼𐠽𐠾𐠿𐡀𐡁𐡂𐡃𐡄𐡅𐡆𐡇𐡈𐡉𐡊𐡋𐡌𐡍𐡎𐡏𐡐𐡑𐡒𐡓𐡔𐡕𐡖𐡗𐡘𐡙𐡚𐡛𐡜𐡝𐡞𐡟𐡠𐡡𐡢𐡣𐡤𐡥𐡦𐡧𐡨𐡩𐡪𐡫𐡬𐡭𐡮𐡯𐡰𐡱𐡲𐡳𐡴𐡵𐡶𐡷𐡸𐡹𐡺𐡻𐡼𐡽𐡾𐡿𐢀𐢁𐢂𐢃𐢄𐢅𐢆𐢇𐢈𐢉𐢊𐢋𐢌𐢍𐢎𐢏𐢐𐢑𐢒𐢓𐢔𐢕𐢖𐢗𐢘𐢙𐢚𐢛𐢜𐢝𐢞𐢟𐢠𐢡𐢢𐢣𐢤𐢥𐢦𐢧𐢨𐢩𐢪𐢫𐢬𐢭𐢮𐢯𐢰𐢱𐢲𐢳𐢴𐢵𐢶𐢷𐢸𐢹𐢺𐢻𐢼𐢽𐢾𐢿𐣀𐣁𐣂𐣃𐣄𐣅𐣆𐣇𐣈𐣉𐣊𐣋𐣌𐣍𐣎𐣏𐣐𐣑𐣒𐣓𐣔𐣕𐣖𐣗𐣘𐣙𐣚𐣛𐣜𐣝𐣞𐣟𐣠𐣡𐣢𐣣𐣤𐣥𐣦𐣧𐣨𐣩𐣪𐣫𐣬𐣭𐣮𐣯𐣰𐣱𐣲𐣳𐣴𐣵𐣶𐣷𐣸𐣹𐣺𐣻𐣼𐣽𐣾𐣿𐤀𐤁𐤂𐤃𐤄𐤅𐤆𐤇𐤈𐤉𐤊𐤋𐤌𐤍𐤎𐤏𐤐𐤑𐤒𐤓𐤔𐤕𐤖𐤗𐤘𐤙𐤚𐤛𐤜𐤝𐤞𐤟𐤠𐤡𐤢𐤣𐤤𐤥𐤦𐤧𐤨𐤩𐤪𐤫𐤬𐤭𐤮𐤯𐤰𐤱𐤲𐤳𐤴𐤵𐤶𐤷𐤸𐤹𐤺𐤻𐤼𐤽𐤾𐤿𐥀𐥁𐥂𐥃𐥄𐥅𐥆𐥇𐥈𐥉𐥊𐥋𐥌𐥍𐥎𐥏𐥐𐥑𐥒𐥓𐥔𐥕𐥖𐥗𐥘𐥙𐥚𐥛𐥜𐥝𐥞𐥟𐥠𐥡𐥢𐥣𐥤𐥥𐥦𐥧𐥨𐥩𐥪𐥫𐥬𐥭𐥮𐥯𐥰𐥱𐥲𐥳𐥴𐥵𐥶𐥷𐥸𐥹𐥺𐥻𐥼𐥽𐥾𐥿𐦀𐦁𐦂𐦃𐦄𐦅𐦆𐦇𐦈𐦉𐦊𐦋𐦌𐦍𐦎𐦏𐦐𐦑𐦒𐦓𐦔𐦕𐦖𐦗𐦘𐦙𐦚𐦛𐦜𐦝𐦞𐦟𐦠𐦡𐦢𐦣𐦤𐦥𐦦𐦧𐦨𐦩𐦪𐦫𐦬𐦭𐦮𐦯𐦰𐦱𐦲𐦳𐦴𐦵𐦶𐦷𐦸𐦹𐦺𐦻𐦼𐦽𐦾𐦿𐧀𐧁𐧂𐧃𐧄𐧅𐧆𐧇𐧈𐧉𐧊𐧋𐧌𐧍𐧎𐧏𐧐𐧑𐧒𐧓𐧔𐧕𐧖𐧗𐧘𐧙𐧚𐧛𐧜𐧝𐧞𐧟𐧠𐧡𐧢𐧣𐧤𐧥𐧦𐧧𐧨𐧩𐧪𐧫𐧬𐧭𐧮𐧯𐧰𐧱𐧲𐧳𐧴𐧵𐧶𐧷𐧸𐧹𐧺𐧻𐧼𐧽𐧾𐧿𐨀𐨁𐨂𐨃𐨄𐨅𐨆𐨇𐨈𐨉𐨊𐨋𐨌𐨍𐨎𐨏𐨐𐨑𐨒𐨓𐨔𐨕𐨖𐨗𐨘𐨙𐨚𐨛𐨜𐨝𐨞𐨟𐨠𐨡𐨢𐨣𐨤𐨥𐨦𐨧𐨨𐨩𐨪𐨫𐨬𐨭𐨮𐨯𐨰𐨱𐨲𐨳𐨴𐨵𐨶𐨷𐨹𐨺𐨸𐨻𐨼𐨽𐨾𐨿𐩀𐩁𐩂𐩃𐩄𐩅𐩆𐩇𐩈𐩉𐩊𐩋𐩌𐩍𐩎𐩏𐩐𐩑𐩒𐩓𐩔𐩕𐩖𐩗𐩘𐩙𐩚𐩛𐩜𐩝𐩞𐩟𐩠𐩡𐩢𐩣𐩤𐩥𐩦𐩧𐩨𐩩𐩪𐩫𐩬𐩭𐩮𐩯𐩰𐩱𐩲𐩳𐩴𐩵𐩶𐩷𐩸𐩹𐩺𐩻𐩼𐩽𐩾𐩿𐪀𐪁𐪂𐪃𐪄𐪅𐪆𐪇𐪈𐪉𐪊𐪋𐪌𐪍𐪎𐪏𐪐𐪑𐪒𐪓𐪔𐪕𐪖𐪗𐪘𐪙𐪚𐪛𐪜𐪝𐪞𐪟𐪠𐪡𐪢𐪣𐪤𐪥𐪦𐪧𐪨𐪩𐪪𐪫𐪬𐪭𐪮𐪯𐪰𐪱𐪲𐪳𐪴𐪵𐪶𐪷𐪸𐪹𐪺𐪻𐪼𐪽𐪾𐪿𐫀𐫁𐫂𐫃𐫄𐫅𐫆𐫇𐫈𐫉𐫊𐫋𐫌𐫍𐫎𐫏𐫐𐫑𐫒𐫓𐫔𐫕𐫖𐫗𐫘𐫙𐫚𐫛𐫜𐫝𐫞𐫟𐫠𐫡𐫢𐫣𐫤𐫦𐫥𐫧𐫨𐫩𐫪𐫫𐫬𐫭𐫮𐫯𐫰𐫱𐫲𐫳𐫴𐫵𐫶𐫷𐫸𐫹𐫺𐫻𐫼𐫽𐫾𐫿𐬀𐬁𐬂𐬃𐬄𐬅𐬆𐬇𐬈𐬉𐬊𐬋𐬌𐬍𐬎𐬏𐬐𐬑𐬒𐬓𐬔𐬕𐬖𐬗𐬘𐬙𐬚𐬛𐬜𐬝𐬞𐬟𐬠𐬡𐬢𐬣𐬤𐬥𐬦𐬧𐬨𐬩𐬪𐬫𐬬𐬭𐬮𐬯𐬰𐬱𐬲𐬳𐬴𐬵𐬶𐬷𐬸𐬹𐬺𐬻𐬼𐬽𐬾𐬿𐭀𐭁𐭂𐭃𐭄𐭅𐭆𐭇𐭈𐭉𐭊𐭋𐭌𐭍𐭎𐭏𐭐𐭑𐭒𐭓𐭔𐭕𐭖𐭗𐭘𐭙𐭚𐭛𐭜𐭝𐭞𐭟𐭠𐭡𐭢𐭣𐭤𐭥𐭦𐭧𐭨𐭩𐭪𐭫𐭬𐭭𐭮𐭯𐭰𐭱𐭲𐭳𐭴𐭵𐭶𐭷𐭸𐭹𐭺𐭻𐭼𐭽𐭾𐭿𐮀𐮁𐮂𐮃𐮄𐮅𐮆𐮇

```
67017c0ea5cb33b3e6967453986e1450b35ad47861f679cf7db5a6c170bcfb67544983ec1e36b27ee
8c5721da39d27dbfa0cdc15ba3cbaa425e8a8b96b81ab665f3ebc41563a0e9270695d3d68887cfab2
c07b290718307f764afba684b17fcfd71323f64206e5fa378b4ee89e80885733080065dd34a5c8388
98906b8d43de9f1d8eb6922bad
<hr>
<address>Apache Server at 127.0.0.1 Port 5243</address>
</body>
</html>
```

This string is 512 characters (256 bytes). Shortly after the discovery of this string the onion went offline...

...until January 29 at 00:05 GMT, when it came back online. This time it contained no HTML. The sole content of the page was a signed message. The content of the message was a hex string (with newlines):

<https://infotomb.com/vnq3e.txt>

This string began with the bytes

```
0x1F 0x8B
```

indicating the string was a gzip file. Converting to binary

```
xxd -p -r onion4.hex onion4.gz
```

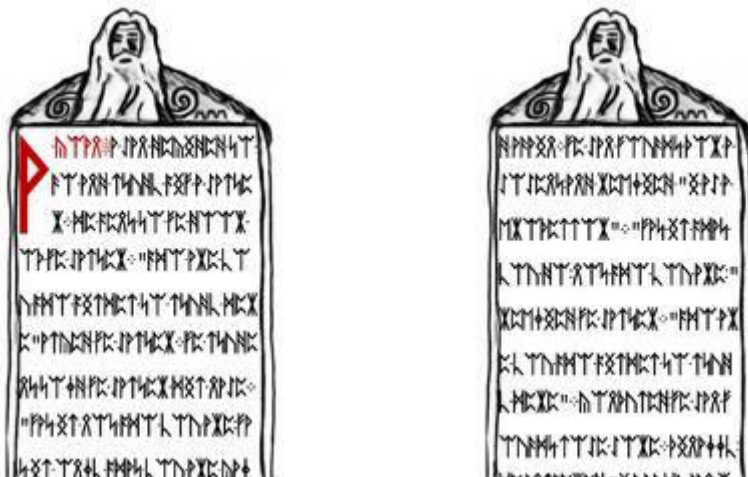
using file to confirm type

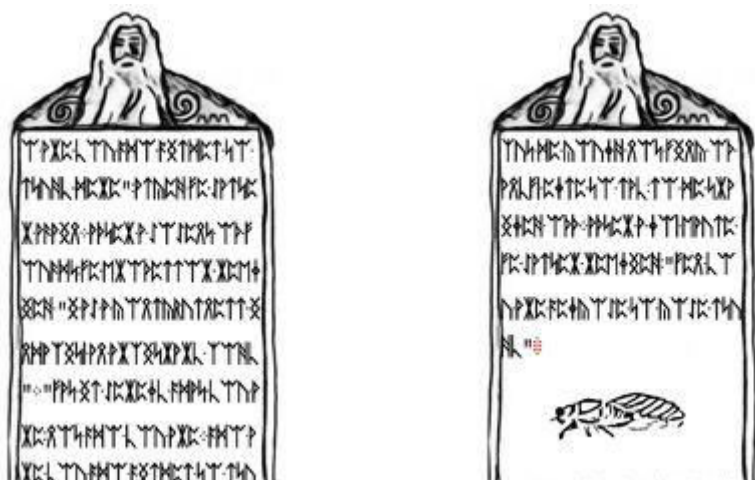
```
$ file onion4.gz
onion4.gz: gzip compressed data, was "data.out", from Unix, last modified: Fri
Jan 24 15:10:12 2014
```

and unzipping

```
gzip -d onion4.gz
```

we are left with binary data containing four images:





Using a similar substitution to the one used in Onion 2

Gematria Primus

Substitution key for deciphering onion 4 koan

f	F	↔	HL	h
h	U	↔	n	q
t	TH	↔	j	d
p	O	↔	EA	l
k	R	↔	OI/VI	x
L	C/K	↔	A	q
X	G	↔	AE	i
p	W	↔	V	d
H	H	↔	D	M
s	N	↔	DO	W
I	I	↔	ONI/DN	X
k	J	↔	T	d
Z	EO	↔	W	M
E	P	↔	g	U
T	X	↔	B	e
h	S/Z	↔	L	d
t	T	↔	Z/S	h
p	B	↔	X	l
m	E	↔	d	d
M	M	↔	OE	z

and applying that to the runes in each image, we obtain a koan.

A KOAN

A MAN DECIDED TO GO AND STUDY WITH A MASTER.
 HE WENT TO THE DOOR OF THE MASTER
 "WHO ARE YOU WHO WISHES TO STUDY HERE?"
 ASKED THE MASTER.
 THE STUDENT TOLD THE MASTER HIS NAME.
 "THAT IS NOT WHO YOU ARE THAT IS ONLY WHAT YOU ARE CALLED.
 WHO ARE YOU WHO WISHES TO STUDY HERE?" HE ASKED
 AGAIN.
 THE MAN THOUGHT FOR A MOMENT, AND REPLIED
 "I AM A PROFESSOR."
 "THAT IS WHAT YOU DO, NOT WHO YOU ARE"

REPLIED THE MASTER. "WHO ARE
YOU WHO WISHES TO STUDY HERE?"
CONFUSED, THE MAN THOUGHT SOME MORE.
FINALLY, HE ANSWERED, "I AM A HUMAN BEING."
"THAT IS ONLY YOUR SPECIES, NOT WHO YOU ARE.
WHO ARE YOU WHO WISHES TO STUDY HERE?"
ASKED THE MASTER AGAIN.
AFTER A MOMENT OF THOUGHT, THE PROFESSOR REPLIED
"I AM A CONSCIOUSNESS INHABITING AN ARBITRARY BODY."
"THAT IS MERELY WHAT YOU ARE NOT WHO YOU ARE"
WHO ARE YOU WHO WISHES TO STUDY HERE?"
THE MAN WAS GETTING IRRITATED. "I AM," HE STARTED,
BUT HE COULD NOT THINK OF ANYTHING ELSE TO SAY,
SO HE TRAILED OFF. AFTER A LONG PAUSE THE MASTER REPLIED
"THEN YOU ARE WELCOME TO COME STUDY."

AN INSTRUCTION

DO FOUR UNREASONABLE THINGS EACH DAY.

The following image



when run through outguess

```
outguess -r onion4image3.jpg out
```

contains a message

For those who have fallen behind:

TL BE IE OV UT HT RE ID TS EO ST PO SO YR
SL BT II IY T4 DG UQ IM NU 44 2I 15 33 9M

Good luck.

3301

This ciphertext was found to be encoded with a columnar transposition cipher. Period = 7, key = 1736254.

Cleartext:

```
TOBELIEVETRUTHISTODESTROYPOSSIBILITYQ4UTGDI2N4M4UIM59133
```

Which of course translates to

```
TO BELIEVE TRUTH IS TO DESTROY POSSIBILITY Q4UTGDI2N4M4UIM59133
```

Formatting that last bit as an onion url yields

```
q4utgdi2n4m4uim.onion Purpose of 9133 at the end of the string was not  
discovered.
```

bringing us to

The Fifth Onion

The fifth onion contained the following (signed) message

<https://infotomb.com/ooxyo.txt> Interconnectedness.mp3 can be found [here](#).

Shortly afterwards it went offline.

Converting the hex string to binary

```
xxd -p -r onion5.hex onion5.bin
```

and checking with file

```
file onion5.bin
```

tells us that it is an mp3 file. Renaming as such

```
mv onion5.bin onion5.mp3
```

and reading the ID3 tags

```
id3v2 -l onion5.mp3
```

shows

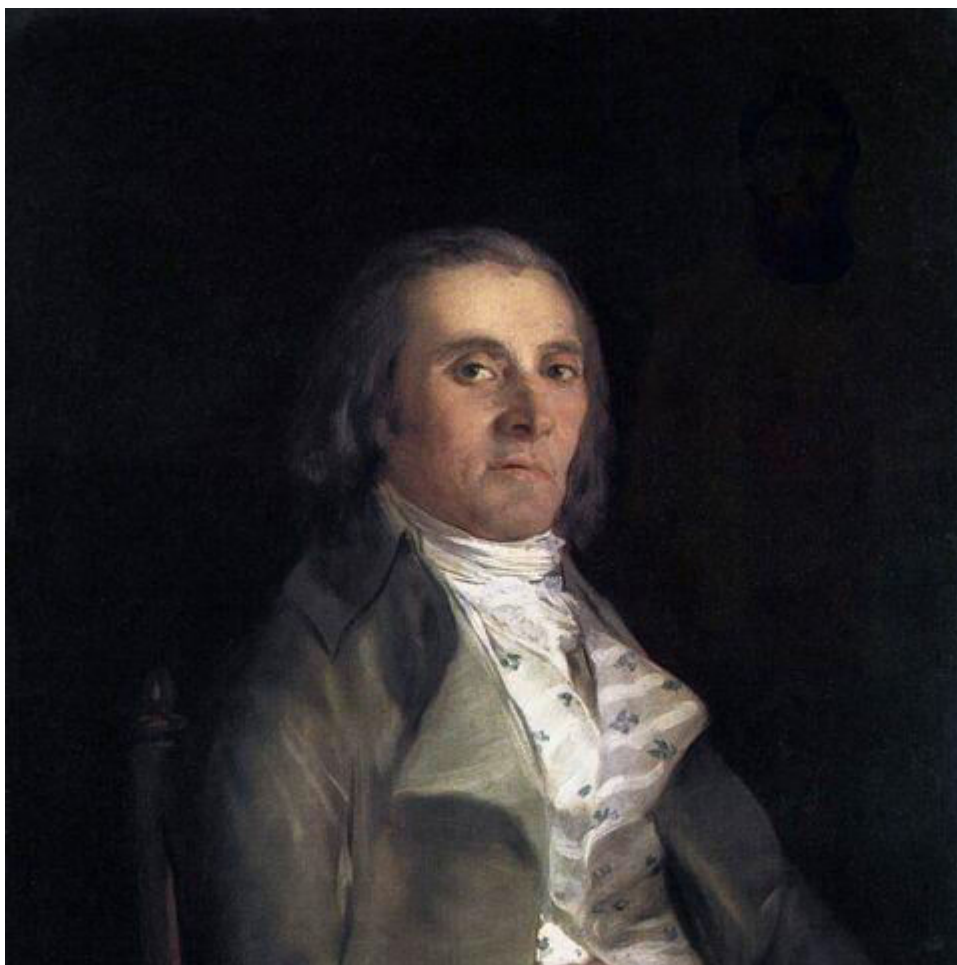
```
id3v2 tag info for his5u.mp3:  
TIT2 (Title/songname/content description): Interconnectedness
```

```
TPE1 (Lead performer(s)/Soloist(s)): 3301  
his5u.mp3: No ID3v1 tag
```

Thus the name of the song is 'Interconnectedness', and the artist is '3301'.

Translating each character of the word 'Interconnectedness' to its corresponding numerical value with the Gematria Primus 2013 and summing yields 772 (a prime). The song itself is 277.133 seconds long.

On January 31, Onion 5 came back online. This time it contained an image





This painting is *Portrait of Andrés del Peral* by Goya y Lucientes. The significance of this is indeterminate.

A subimposed image of a man can be seen in the upper right corner. Adjusting image filters reveals hidden information:

--Image temporarily omitted due to wikia server maintenance--

The man in the upper right is supposed to be Grigori Rasputin. The significance of this is indeterminate. The two columns of numbers are (left to right)

181
7
15
16
966
456
351
7

which sums to 1033, and

```
966
1071
626
204
434
```

which sums to 3301. The significance of these numbers is indeterminate.

Applying outguess to the original portrait

```
outguess -r onion5portrait.jpg onion5portrait.outguess
```

and checking the output with file

```
file onion5portrait.outguess
```

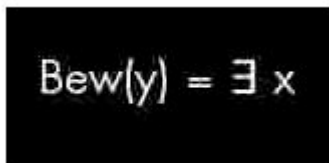
indicates that the retrieved data is a bzip compressed file. Decompressing

```
bzip2 -d onion5portrait.outguess
```

gives us a text file with a signed message.

<https://infotomb.com/772hf.txt>

Breaking up the three hex blocks in the message and using xxd to convert to binary yields two JPG images and an MP3 file.



The equation in the first image is part of the **Godel** incompleteness theorem. The mp3 file is a segment of **Bach's** Trio Sonata in G Major (BWV 1039). ([Some say](#) that it is BWV 1033) The picture of the eye is a painting by M.C. **Escher** called 'Eye', painted in 1946.

The last bit of the signed message

```
3PI:6:1:3
LML:1:1:1
3
ETOATS:19:9:1
...AF:5:3:1
AMO:13:10:1
CC:8:6:1
CBIA:3:7:2
CFAF:5:23:6
SPR:1:8:1
7
C[1]:4:5:3
AWDV:6:2:1
C[2]:2:17:5
SC:3:17:1
AOGS:2:8:1
ONION
```

is a book code. The key is *Gödel, Escher, Bach: An Eternal Golden Braid* by Douglas Hofstadter. The format of the code is

```
<chapter>:<line>:<word>:<letter>
```

Applying the code to the book:

3PI:6:1:3	Three-Part Invention 29	(u)
LML:1:1:1	Little Harmonic Labyrinth 103	(t)
3		(3)
ETOATS:19:9:1	Edifying Thoughts of a Tobacco Smoker 480	(q)
...AF:5:3:1	... Ant Fugue 311	(t)
AMO:13:10:1	Introduction: A Musico-Logical Offering 3	(z)
CC:8:6:1	Crab Canon 199	(b)
CBIA:3:7:2	Canon by Intervallic Augmentation 153	(r)
CFAF:5:23:6	Chromatic Fantasy, And Feud 177	(v)
SPR:1:8:1	Six-Part Ricercar 720	(s)
7		(7)
C[1]:4:5:3	Contracrostipunctus 75	(d)
AWDV:6:2:1	Aria with Diverse Variations 391	(t)
C[2]:2:17:5	Contrafactus 633	(v)
SC:3:17:1	Sloth Canon 681	(z)
AOGS:2:8:1	Air on G's String 431	(p)
ONION		

yields

```
ut3qtzbrvs7dtvzpONION
```

and formatting correctly yields

```
ut3qtzbrvs7dtvzp.onion
```

bringing us to

The Sixth Onion

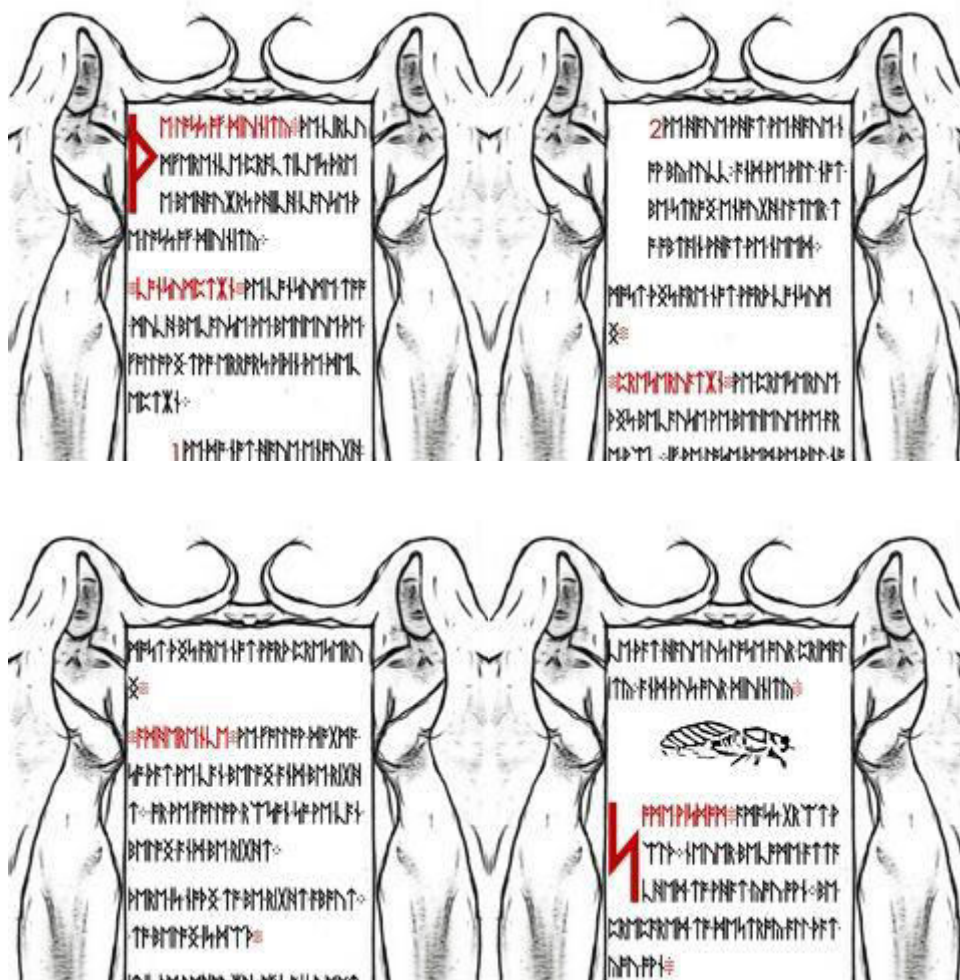
ut3qztzbrvs7dtvzp.onion

The first thing found at this onion address was a large block of hex after the HTML comment

The hex was

<https://infotomb.com/hrz8z.txt>

It contained four JPEG images in sequential, non-reversed order. They are



The runes were not enciphered, and they read:

Page 1:

THE LOSS OF DIVINITY: THE CIRCU
MFERENCE PRACTICES THRE
E BEHAVIORS WHICH CAUSE TH
E LOSS OF DIVINITY.

CONSUMPTION: WE CONSUME TOO
MUCH BECAUSE WE BELEIVE THE

FOLLOWING TWO ERRORS WITHIN THE DECEPTION.

1 WE DO NOT HAVE ENOUGH
OR THERE IS NOT ENOUGH

Page 2

2 WE HAVE WHAT WE HAVE NOW BY LUCK, AND WE WILL NOT BE STRONG ENOUGH LATER TO OBTAIN WHAT WE NEED.

MOST THINGS ARE NOT WORTH CONSUMING:

PRESERVATION: WE PRESERVE THINGS BECAUSE WE BELIEVE WE ARE WEAK. IF WE LOSE THEM WE WILL NOT BE STRONG ENOUGH TO GAIN THEM AGAIN. THIS IS THE DECEPTION.

Page 3

MOST THINGS ARE NOT WORTH PRESERVING:

ADHERENCE: WE FOLLOW DOGMA SO THAT WE CAN BELONG AND BE RIGHT. OR WE FOLLOW REASON SO WE CAN BELONG AND BE RIGHT.

THERE IS NOTHING TO BE RIGHT ABOUT. TO BELONG IS DEATH.

IT IS THE BEHAVIORS OF CONSUMPTION, PRESERVATION, AND ADHERENCE

Page 4

CE THAT HAVE US LOSE OUR PRIMALITY AND THUS OUR DIVINITY:

SOME WISDOM: AMASS GREAT WEALTH. NEVER BECOME ATTACHED TO WHAT YOU OWN. BE PREPARED TO DESTROY ALL THAT YOU OWN:

AN INSTRUCTION: PROGRAM YOUR MIND. PROGRAM REALITY

Each of these images contained a PGP signed outguess message. They're all essentially the same, following the form of

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

Create one Tor hidden service that can accept CGI file uploads.

When this hidden service returns and can accept input, post the three magic squares and the URL to your Tor hidden service here.

Work alone.

```
33333333333333333333
310      12      103
3          3
312      14      123
3          3
310      12      103
33333333333333333333
```

Good luck.

3301

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

iQIcBAEBAGAGBQJS24E0AAoJEBgfAeV6NQkPNPEP/jxtRsM2AOE3KRChpl1IHxGe
oMyd/YjXW7/o8X6Cv+AYhzQhRhqOQPL0N+fVC2WNO64CGnOmTLbhZPoXpV1giSmA
UguBIWZ59MmitGVmiz68M/i5H68h7s7eXoC5u7/iZjVMQBr86J1iJyeabVjJMSp+
OI8ouTuVAZ5lccvUy9UpV82GtGZoM7P1xxWJGpM3LIz6mv7VdfogrCNAW0jhZ2/x
8eiiHFuB5oktc9uTbIqhJQsESuc1u/uMkeb1OGXovKmD+zLtq+DPbWo8P0lnT70V
pyLOLM5CWsAjb1U+5ohK57yFP6AV6x4l97BQyRmMOojh35QGKPVULZG4sRPKsuG5
nw93gRi6/eQ/aQvQuEvkf4lbj/V5G4koj/YcQhyjAWdo1UP19zkUXs2lKH7sUUms
P1WV6eyL6rAqpUXfwpDSfaPTPquIwuFsEl5z/dl4IXnR3s+LQjlFDO3DE2d9Q1Er
h5daiLFevH+wyoJ5aPOsSkT+QJqCrVQNnbbQYzYKeMKAshu1LWuk1ZQ0XAEA6C2b
zbiPcXg0OMO+VWkhscZwxIHr1N5TVDj3N0szCfUe7lrYZhE0F/TL50NkGxw9+2qH
byDA8E4Yhe2c7pUVgs3OQLX46N4S0lbsH2MNXO9Y5jjI2Oj+OwLQ07F8jouNvXN4
kY3+nCV1PPLtiOulCCP1
=/MnD
-----END PGP SIGNATURE-----

Note the number square (not magic) in the middle of the message:

```
10      12      10
12      14      12
10      12      10
```

The only difference between the messages in each image was the number used to make the border of the number square. Page 1's outguess (above) used '3', page 2 used '3', page 3 used '0', and page 4 used '1', spelling out 3301.

The onion went offline shortly afterwards. Six days later it came back online, sporting the following message:

-----BEGIN PGP SIGNED MESSAGE-----

Hash: SHA1

Hello. You have done well to come this far.

Please paste the magic squares into the appropriate textareas below, then provide the URL to your Tor hidden service.

The path to your CGI script which accepts uploads should be '/cgi-bin/upload' and the HTML form input which accepts file uploads should be named 'file'.

Additionally, please generate a GnuPG key pair, and place the public key in the location '/key.asc'.

We will contact you soon.

Good luck.

3301

-----BEGIN PGP SIGNATURE-----

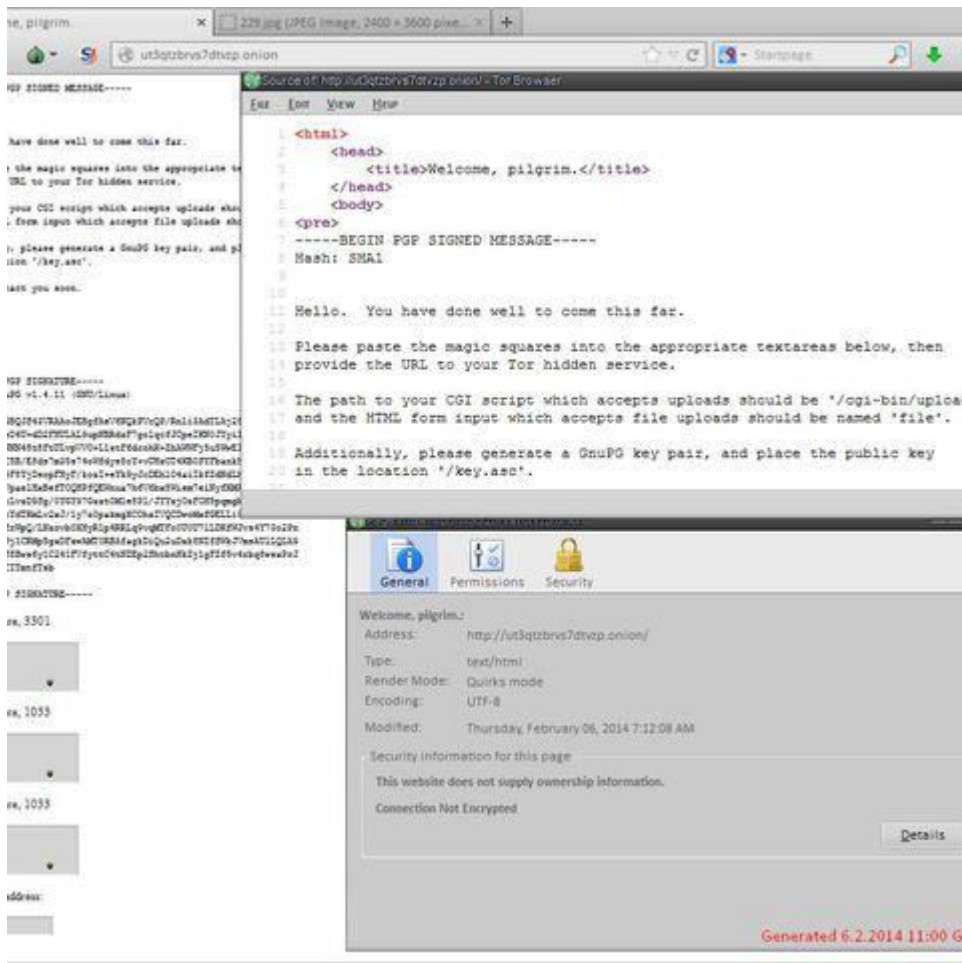
Version: GnuPG v1.4.11 (GNU/Linux)

iQIcBAEBAgAGBQJS43VRAAoJEBgfAeV6NQkPVcQP/Rnli3AdTLAj28WlSMHTD6v0
Q67n89uGF6ZeD4U+dD2FHULAL9upNBRdzF7golqcFJCpeIKN0JYyilpGgSyTQmx+
yJXinlq4ZY+NNN45t8FtULvpVVO+L1ztF6dcohK+ZhAWWFj5u5WwEINx0mo+TE35
S7imfprBdk2C5B/E8ds7m35s74oWfdys8oY+vUHZOT4KB0SYFbankH6aLIe7fiTa
STB1Effelhg9F8YjDsopFHyF/kozI+eYk9yJcDEh1O4aiIkfZdNdLhXz80SIKw9v
ryjTCPUJfFrVpaelHxBefTOQHPfQEWnua7h6V6bx8Wiem7eiNyfXMAkluoIU9zWW
FbA+MIFZ711kLvzD9Sg/0YGY97Gzzt0M1e8Pl/JYYzjOzFOH5pqmgMoOTBO0bvV+
d+QaLLiKmH0cYdTKmLv2xJ/1y7z0pakmgXCOhzIVQCDwoMxfGELLi9MNroaZFK3e
JzeDy828EfafrWpQ/LNzovb0XHyR1p4RRLq9vqMTFo0U0U71LDKfWJvs4Y73o2Pn
QRNzn2+2GPQPj1CRMp5gxDFzwAMT0RBAfagkDiQu2uDxk8NZfSWkJVmsAU1LQLA9
7Wa5zuxPNvBf8Bws6y1C241FVfyttC4tNZEp2ShtbnHkZj1gFZf5v4rbq8wsxPrJ
Jp8kkuLi0PWCITmtfTsb
=TwC1

-----END PGP SIGNATURE-----

and then three text boxes where the referenced 'magic squares' had to be pasted. At the bottom was a text box for the onion url of the hidden service mentioned in the previous message.

Screenshot of the page:



The page accepted any magic square that matched the criteria (e.g. first box square had to be order 5 and sum to 3301). Some time after this, it was discovered that the mp3 'interconnectedness' found on Onion 5 contained steganographically hidden data. The tool used to hide this data was 'OpenPuff'. The data hidden was a text file named 'magicsquares.txt' containing the following:

434	1311	312	278	966		
204	812	934	280	1071		
626	620	809	620	626		
1071	280	934	812	204		
966	278	312	1311	434		
7	375	236	190	27	17	181
351	223	14	47	293	98	7
456	232	121	114	72	23	15
16	65	270	331	270	65	16
15	23	72	114	121	232	456

7	98	293	47	14	223	351
181	17	27	190	236	375	7
272	138	341	131	151		
366	199	130	320	18		
226	245	91	245	226		
18	320	130	199	366		
151	131	341	138	272		

Two order 5 and one order 7 magic square, summing to 3301, 1033, and 1033.

Upon submitting these squares and a link to your hidden service, the following page displayed:

```
< html >< head >< /head >< body >
< p > Thank you for you submission. < /p >
< img src="[view-source:http://ut3qtzbrvs7dtvzp.onion/107.jpg /107.jpg]" / >< br
/ >
< img src="[view-source:http://ut3qtzbrvs7dtvzp.onion/167.jpg /167.jpg]" / >< br
/ >
< img src="[view-source:http://ut3qtzbrvs7dtvzp.onion/229.jpg /229.jpg]" / >< br
/ >
```

Note that the <html> tag is not properly closed.

The images on the page:



434	1311	312	278	966
204	812	934	280	1071
626	620	809	620	626

The final image is enormous, perhaps indicating the end of the book.

Decoding the runes on the page using a Vignere cipher and key CIRCUMFERENCE yields

```
A KOAN: DURING A LESSON: THE MASTER EXPLAINED THE I: "THE I IS THE VOICE OF THE CIRCUMFERENCE," HE SAID. WHEN ASKED BY A STUDENT TO EXPLAIN WHAT THAT MEANT, THE MASTER SAID "IT IS A VOICE INSIDE YOUR HEAD." "I DON'T HAVE A VOICE IN MY HEAD," THOUGHT THE STUDENT, AND HE RAISED HIS HAND TO TELL THE MASTER. THE MASTER STOP
```

--page change--

```
PED THE STUDENT, AND SAID "THE VOICE THAT JUST SAID YOU HAVE NO VOICE IN YOUR HEAD, IS THE I." AND THE STUDENTS WERE ENLIGHTENED:
```

The text on the final page is not encoded. Transcription via Gematrius Primus yields:

```
AN INSTRUCTION: QUESTION ALL THINGS: DISCOVER TRUTH INSIDE YOURSELF: FOLLOW YOUR TRUTH: IMPOSE NOTHING ON OTHERS.
```

```
KNOW THIS:  
434 1311 312 278 966  
204 812 934 280 1071  
626 620 809 620 626  
1071 280 934 812 204  
966 278 312 1311 434
```

The End

As usual, after the final submission we heard nothing. The IRC channels slowly died. The onions all went offline.

Author's Note: when I say we heard nothing, 'we' references the collective of solvers. Cicada does not recruit groups, only individuals.

Long after the community as a whole had given up, a response was finally received. Some got a message similar to the one from the end of 2013 (*or so they claim*), and the rest got a link to an onion, containing the remaining pages of the Liber Primus, a 58 page dump of images. These will be discussed [here](#).

Infographics

Schematic presentation of all steps of 2014 puzzle: <http://imgur.com/42VKTWB>

v2

2014 puzzle scheme

