

vagrant

```

File size          1675 bytes (last modified 2019-09-28 08:21:51 +0200 CEST)
File Sha256        c95842bf221d67a85a26796ba13fb58951985bffc01f399fa0d133b89fa08a52
Base64 Sha256      2cb2793d35f0d2d6d7cc2b5c31aaadf32dd07543ba9da3239366dd3b729024cc
GPG AES256 password 8B0ogB+>T9zJUGu)

```

This is the paper backup of vagrant. The original file has been compressed using *gzip* then encrypted with a *gpg* compatible *AES256* cipher and finally *base64* encoded. The *base64* stream has been split into 4 blocks. For each block its QR-code and textual representations provided along with the block stream *sha256* sum (one line without any blank character). The *base64* encoded file can be restored by using any QR-code scanner or OCR tool. Yet the content can be manually typed. To help you each line can be verified using a *IEEE-CRC32* checksum. As for the block *sha256* checksum, the whole *base64* stream checksum has been computed without any blank character.

To compute a *IEEE-CRC32* sum:

```
echo -n line | gzip | tail -c 8 | hexdump -n 4 -e '"0x%.8x\n"'
```

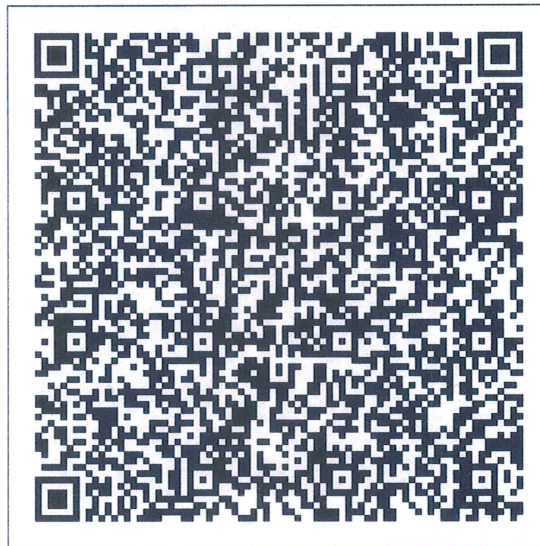
To remove all blanks from the *base64* stream you can run:

```
cat vagrant.b64 | sed 's/ *#.*//;s/ +//;' | tr -d '\n'
```

The file restoration command (depending on your system the *-D* option can be *-d*) from the *base64* file is:

```
cat vagrant.b64 | base64 -D | gpg -d --batch --passphrase '8B0ogB+>T9zJUGu)' | zcat > vagrant
```

vagrant starts here



Line	Block 1 (lines 1 to 8) base64 data								CRC32
1	wx4EBwMI	BqkfXuKj	euFgjtGb	bDa0zfkB	6Gz4lUG5	cNzS4AHk	Bxp8hDaC	9p8SziP9	0xa3c411c2
2	0qs3++F2	u+AS4G3h	pw7g50Jt	Mpui4Nnj	i8wtVFM6	m+DgXeGX	UuBx50p4	kn8hobNq	0xf1bb67a8
3	aZc1Ai1D	XMVg8mke	s5QG87Xh	k18DR97d	T9j1S1rV	tAwH5TNY	V3lVG6G0g	N41ht90I	0xecbd6cd3
4	M0cSIoNX	h8A0cPmp	2i9syIMA	ciKgfWbG	4yaFn0Ob	P03GXNta	IkmUTr/q	Ry+H0R2o	0x88be7b31
5	mjlTfQfJ	peD240cp	YxsgocmI	9Jv/gGLk	4P7mBc3A	lEgrnXK5	nWH8Pq5J	H4Y3BX9r	0x1113febf
6	AtGu7Wnq	6zkymvVX	07EhCzqa	INv4EKAS	SPqWmX34	+7IbAiH+	xmK/6UtJ	wuAJ5TqJ	0x1fada7bf
7	vTLLkIEX	8ypkFm4M	/yPEF06V	8+V68dp0	h0g42QPt	4Hbk1C/m	uWVE163T	qUETzLvp	0x89dc0326
8	J+CM5515	WNh2eyEq	IFptuDUM	pHdxFxFE	rX6eEeCp	e8dvZKON	4f1M8cjU	SfW45QfB	0x63f36acf

Sha256 of block 1: 48c67cba9b86d955bec86ec7e5dd1719dd6754358dbe2f9a25dacde4da5f5c7a



Line	Block 2 (lines 9 to 16) base64 data								CRC32
9	8iG7Wqoc	0e0nCKFJ	Etim4u1r	UXB/qB80	spg20TBa	2XW6dGcG	k4dr69am	Vudg0JVg	0xd73492b
10	/hfGvSbG	Mly/Qm1/	g/w69Ueg	+9fr61W7	zo32Uut1	6vbIr+uE	4Mfmw8gB	gMjpzu00	0xac274faa
11	Np2zAv6p	SgB3bZvv	u2yNQpZH	F4uiEUvb	B1SGLPrq	fDv7wtRY	+qxpUuSI	7GPw4HJQ	0x21d8ffef
12	kh97iwWk	1+DF5aXi	h1W0+vEv	KU01mi0U	B2V+AN1+	VymWtx+h	BJXp5oyP	4LzkEYuC	0xf0b03d40
13	VjF5tL9o	qM9rhWrc	0+BS540J	UKrAk5+v	D7jUTI/b	YxHh0rh6	YD6u2Uf+	rwWbw0c5	0x98460f34
14	z6SgSV14	sCptTQFL	mGsy+8Uq	+w1BycDg	DISy3Vhv	2c4TWDCt	lv4NWwqk	jPxGeX06	0x18168c74
15	IhoJpGUM	5T8hfIXU	1YF+OVUd	ceEF3qEm	tyec19Dh	BB0xormk	F5W/e0cj	doSSb7+T	0x5e882dc2
16	4CTmbV2D	//2gsk1g	Y0mKajHu	iobpgJ4e	kJKuDjAD	knrTyp9t	yt1LczLN	IEjBGFdV	0x5ca46b78

Sha256 of block 2: 00d67aea61154a34c718e20f0721715a0541ad6d4cd8a4f63cffbba289755862



Line	Block 3 (lines 17 to 24) base64 data								CRC32
17	yQCO6Nz	4SvF81hS	ipCF1prk	2eAZ5aXy	18R914oE	DvEjS5cI	nVNzoQ4f	NpJPDq9C	0x6a2f6a39
18	1bg/6Y29	4K/kfpLI	F1wg2chJ	oC/10T2U	F+CR555y	CZ9a7gx3	gh8P+L03	jLLiN4na	0x91b06f71
19	VVL/wcnc	WeXTJCyv	OBppU8VW	pgRGbT2E	fe7/amw9	sb/Oj3zE	ZSSWpt5Z	wZYtz5b7	0xd4109d36
20	Cy7chTsj	+PacUzjw	y9GVHcBe	NvCUh069	eIXsibOK	7MTGNbnw	EB/+x1+r	Brj4Ibwk	0x0456fef8
21	c5D3+XA/	4rmiWkmd	4IHmk1/o	BB6+M8GW	TE1AF3X0	geLALglW	Eyvvdzf5	UqhRszZ3	0x075fac96
22	4Z2Ms03b	22ZDP054	/s0XWSEr	M5sg9mZV	U5FaSgqz	cuAA5a1A	G1WZRT3p	ZTeXzJ9H	0x77068e10
23	E0MF1GPS	dKbQ/5DJ	OdFTZ1V8	4DXknSWX	awCgar3z	IZpmY1XC	ZeBZ593S	5MBQ6jWq	0xbe724de2
24	GN7/2ysk	ulqzpPz0	tf87tTRM	w6IcULTC	GsdKetqJ	1nt16aSW	fMKKCK8K	Ashw5U1b	0x2468097f

Sha256 of block 3: 9281bec9e5532bbac12942bc2057270ad0060123f2e18c919e1e73f456b8723b



Line	Block 4 (lines 25 to 31) base64 data										CRC32
25	fMVV8/bb	Ci/0xESN	OWfTcY5/	II6DUp/W	nzKQfi/u	YQ0pV3Kn	yDJNzL+t	59gB2LwN			0x750a8b96
26	M0aqPxc1	W5FmqBKw	i14W7CVX	aZQPN07M	4GXmfCX7	X6GNMye0	x9Ry2lvX	bFCczPxw			0x841a26f6
27	aZh8QuI2	p1kInWgx	pjUp1VD7	qp0Htvqr	Gk0bMaaU	prVUzz2f	fN/h3YCs	9eDm5Wa6			0x2a2cd246
28	bcTlwoF7	gUX5A7n0	q+DbLNdi	/5JbqdXf	bTG/naxE	4ADkLoyh	EVmitq5W	csuWAeKV			0x04e3fc65
29	F+Dk5iOM	GF1vn8v7	PsZB9N4s	MLYRLNgF	Yn1/h3aZ	/jmyKtZ5	FF6JhAlg	dpEaaAJt			0xa42c292c
30	yKT0It2i	+75g2Hib	jmeNVbMP	ChPgcORI	xGpEz/rS	Tu8Vy6yH	2rhI4NTi	BjXNK0AO			0xd408c32b
31	44x0+5AB	n+wI4BPK	GYZD8chE	z2qNCiku	m50rseL0	Qyy/4fhS	AA==				0xb17304ec

Sha256 of block 4: bcdec41ef0d3aa360d03ccb8d102de0f01b64a1a8aaa1fe1adcf5b7c6ea2566e

vagrant ends here