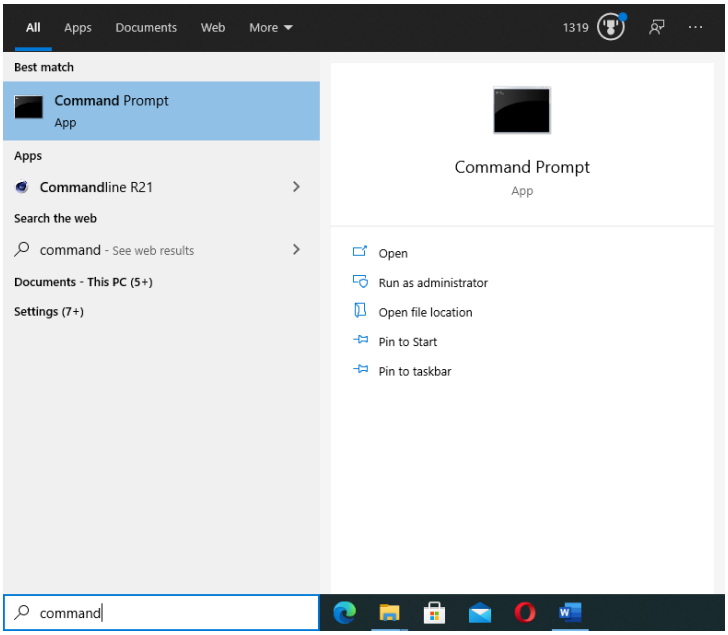
How to Run CryptoKeysTool

1)Download and install the appropriate Java Development Kit for your system if not already installed.

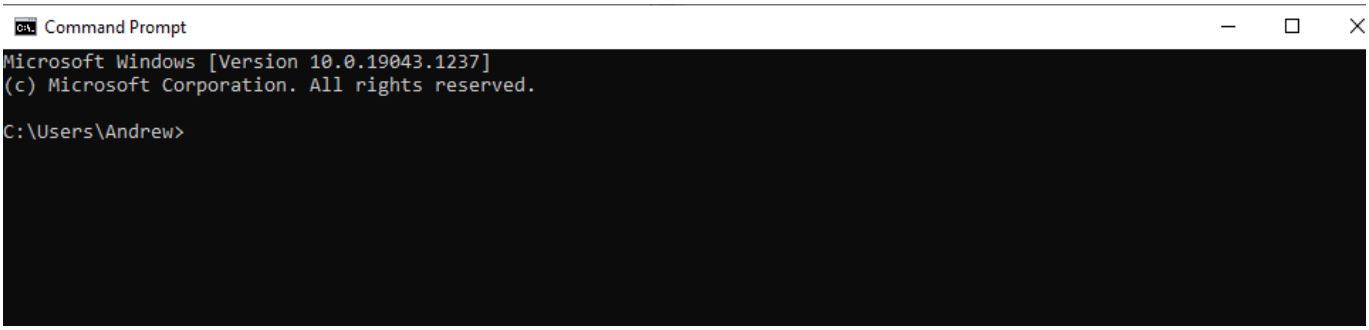
(Note: This has been tested on a Windows 10 64-bit pc. Best to do this operation with a computer disconnected from the internet....keep in mind anytime private keys hit a computer in anyway, you are at risk of losing your crypto associated with that private key from sophisticated hackers. This is why I store all my private keys offline in a non-electronic format. And best practice is to move all money out of any address to a new address once the private keys are used online anywhere. Thus the benefit of this tool. There are more crypto currency addresses than there are atoms in the universe. They are expendable and free to generate.)

2)Download /unzip and save to computer CryptoWebZip from frongello.com. This will contain this pdf, CryptoKeysTool.jar, and CyberChef. Unzip contents and save somewhere on your computer.

3)Open a command prompt- In windows just type "command prompt" in search bar

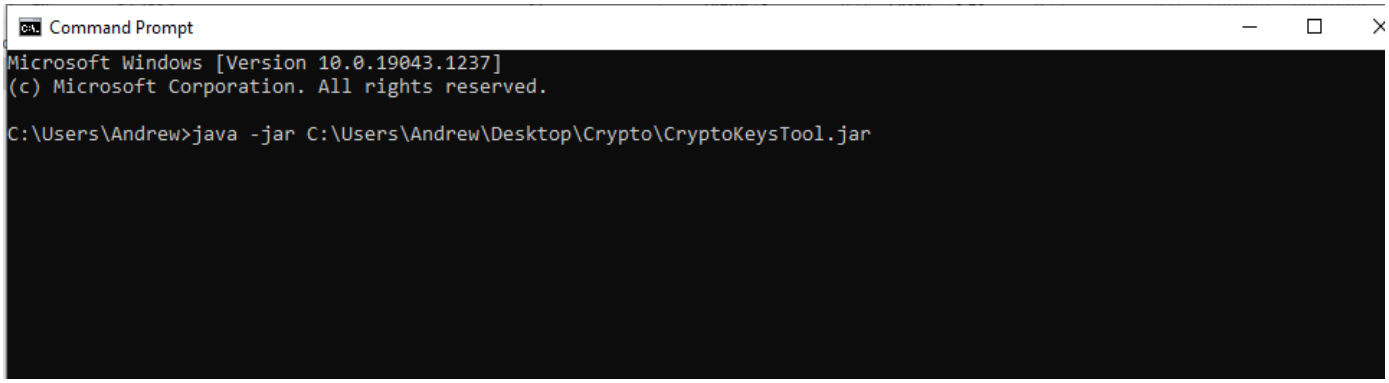


Click on Command Prompt (App)

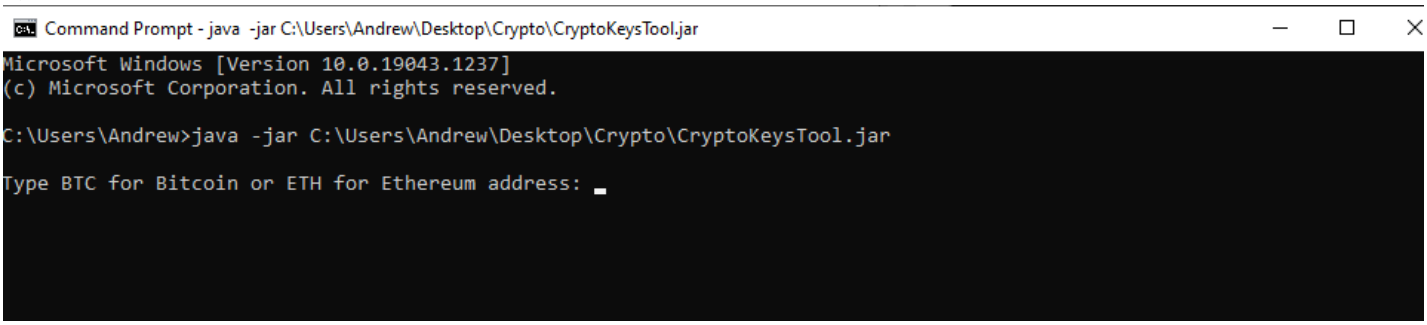


4)On my computer, I saved the CryptoKeysTool to the following location
C:\Users\Andrew\Desktop\Crypto

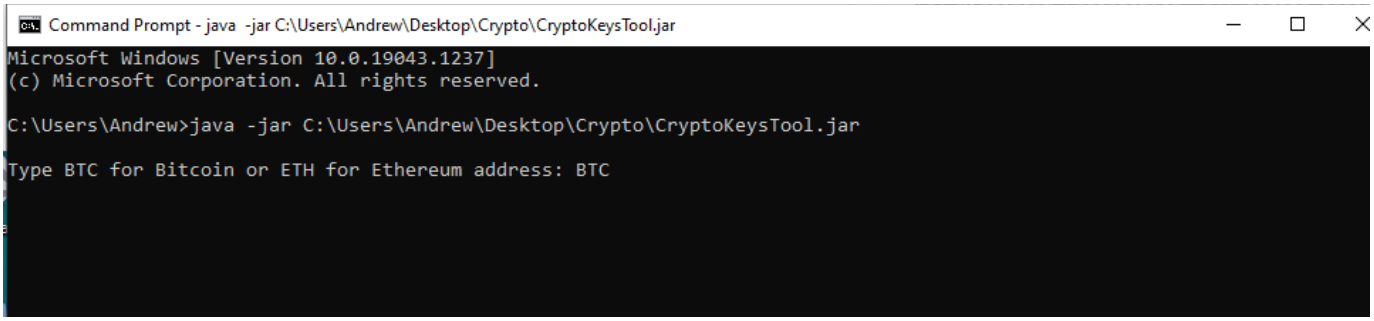
5)Just right of the carrot in the Command prompt I add the following line
"java -jar C:\Users\Andrew\Desktop\Crypto\CryptoKeysTool.jar"
Substitute in your CryptoKeysTool location that you saved to



6)Hit Enter



7)Type in "BTC" or "ETH" depending on which currency you are working with.



8)Hit Enter

```
Command Prompt - java -jar C:\Users\Andrew\Desktop\Crypto\CryptoKeysTool.jar
Microsoft Windows [Version 10.0.19043.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Andrew>java -jar C:\Users\Andrew\Desktop\Crypto\CryptoKeysTool.jar

Type BTC for Bitcoin or ETH for Ethereum address: BTC

You have selected BTC

Type Y if you want to enter custom private key in HEX format, otherwise type N then random private key will be created i
n the app. Y/N ?:
```

9)At this point you can add your own 64 digit random hexidecimal private key or let the program generate one for you. This functionality is here because sometimes you just want to generate a valid set of private key/public key/address and other times you already have the private key and want to generate the public key and address. This tool saves you the requirement to save the entire set.....you can just save the private key and generate the rest as necessary.

10)In this example, we are going to type Y and generate our own 64 digit hexidecimal number.

```
Command Prompt - java -jar C:\Users\Andrew\Desktop\Crypto\CryptoKeysTool.jar
Microsoft Windows [Version 10.0.19043.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Andrew>java -jar C:\Users\Andrew\Desktop\Crypto\CryptoKeysTool.jar

Type BTC for Bitcoin or ETH for Ethereum address: BTC

You have selected BTC

Type Y if you want to enter custom private key in HEX format, otherwise type N then random private key will be created i
n the app. Y/N ?: Y
```

11)Hit Enter

```
Command Prompt - java -jar C:\Users\Andrew\Desktop\Crypto\CryptoKeysTool.jar
Microsoft Windows [Version 10.0.19043.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Andrew>java -jar C:\Users\Andrew\Desktop\Crypto\CryptoKeysTool.jar

Type BTC for Bitcoin or ETH for Ethereum address: BTC

You have selected BTC

Type Y if you want to enter custom private key in HEX format, otherwise type N then random private key will be created i
n the app. Y/N ?: Y

You have selected Yes, Type private key in HEX format:
```

12)Here we need to stop for a quick lesson on HEX or hexadecimal. (You can skip this step if familiar)

We all grew up using the decimal base 10 counting system. There are 10 digits and when we get to the tenth we carry a one over to the left and repeat. We do the same thing in the binary base 2 world. In binary we only have two digits, 0 and 1. This is how computers count...magnetic polarity in the computer (north and south) are interpreted as 0 or 1. Another way to think about the 4 bit binary numbers below (each bit is a 0 or 1....and below we grouped 4 bits together)...is that the first position from right represents a value of 1, second position is a value of 2, third postion represents a value of 4, and fourth position represents a value of 8. And the 0 or 1 indicates whether we include that value in our number. This way we can write 16 different values with the 4 bit binary numbers below. In hexadecimal, we want to represent each of those 16 values with a single character. But we can't do that with 0-9, so after 9 we count up with A, B, C, D, E, F. Then we have the 16 unique characters to represent the 16 different values.

Decimal (Base 10)	Binary (Base 2)	Hexadecimal (Base 16)
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Ethereum and Bitcoin Private Keys are 256 bits long....represented by 64 Hexidecimals.
(Recall:Hexidecimals are 4 bits each)

Example:

2e79357628ef251e7b7d6c73472fb698cc651881bd6221bb8709699eecd723bd

But using the logic/table above.....the hexadecimal number can also be represented as 256 bits.

0010111001111001001101010111011000101000111011110010010100011110011110110111110101101100011100110100001110010111110110110100110001100110001100101000110001000000110111101011000100010000110111011100001110000100101101001100111101110110011010111001000011101111 01

Notice the first 4 characters in the hex

2 = 0010

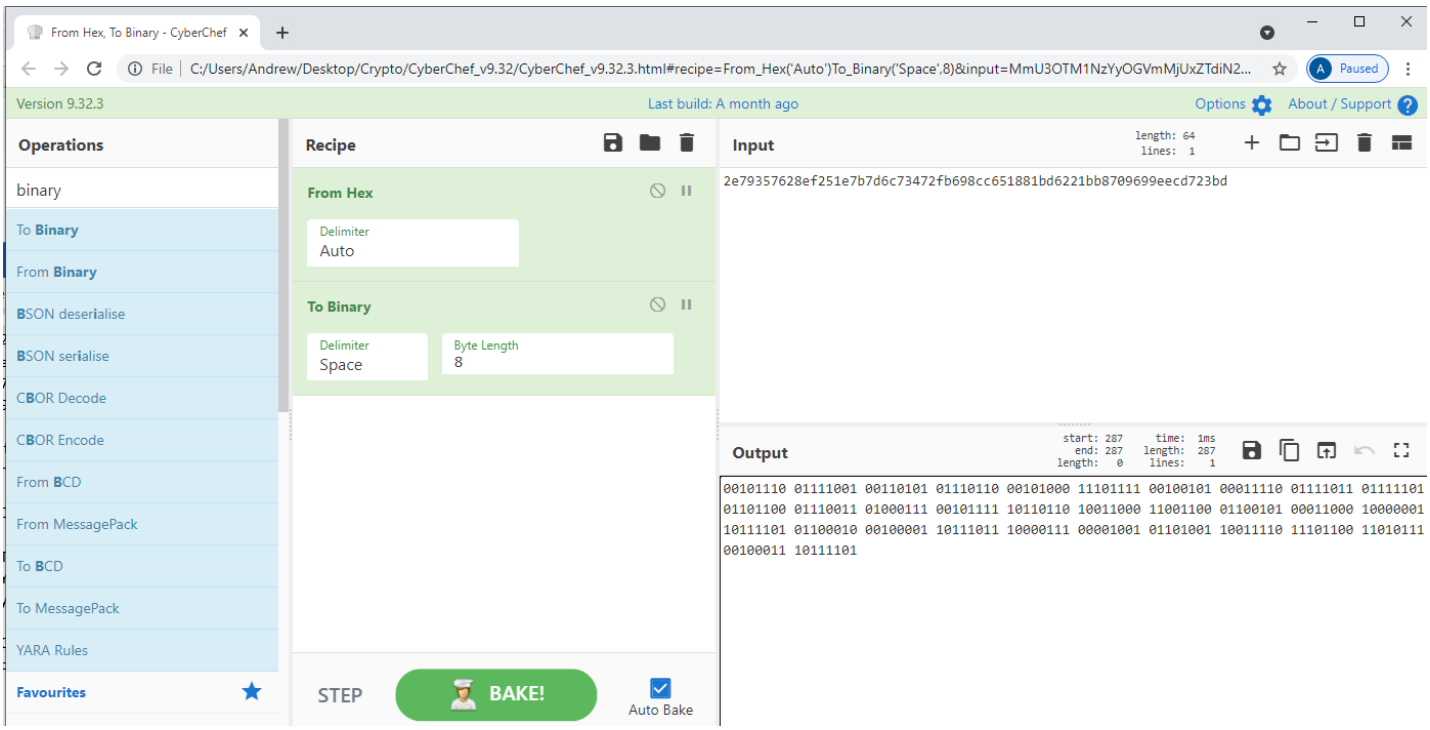
e = 1110
7 = 0111
9 = 1001

if you look carefully you can see that these are the first 16 bits of the 256 bit binary representation of our hexadecimal number.

0010 1110 0111 1001

This is exactly how you convert from hexadecimal to binary and vice versa.
You could have taken 256 random coin flips for example and converted them to 64 hexadecimals and viola....you made your own private key that you can run through this crypto program.

CyberChef is a great tool for easily converting between number formats. This is included in the zip and it can run in a browser offline.



10)In our example we type or copy/paste the hex above into our program and hit enter

```
Command Prompt - java -jar C:\Users\Andrew\Desktop\Crypto\CryptoKeysTool.jar
Microsoft Windows [Version 10.0.19043.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Andrew>java -jar C:\Users\Andrew\Desktop\Crypto\CryptoKeysTool.jar

Type BTC for Bitcoin or ETH for Ethereum address: BTC

You have selected BTC

Type Y if you want to enter custom private key in HEX format, otherwise type N then random private key will be created i
n the app. Y/N ?: Y

You have selected Yes, Type private key in HEX format: 2e79357628ef251e7b7d6c73472fb698cc651881bd6221bb8709699eecd723bd

You have typed private key as HEX: 2e79357628ef251e7b7d6c73472fb698cc651881bd6221bb8709699eecd723bd

Type Y for compressed public key or N for uncompressed Y/N?:
```

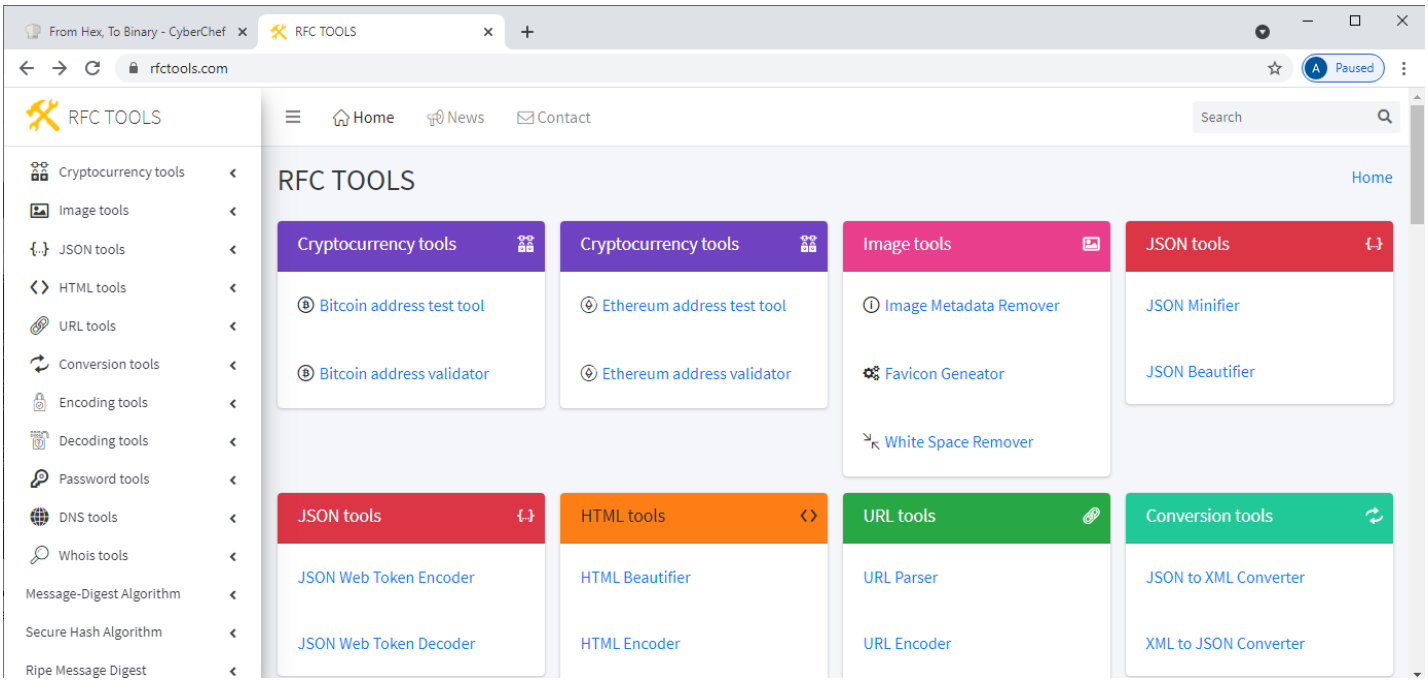
11)Here you select whether you want a compressed or uncompressed public key
(Note: compressed public keys are far more commonly used....but many sites/wallets are backward compatible)

We will type in 'Y' for compressed and hit Enter

12)

```
Command Prompt

Type Y for compressed public key or N for uncompressed Y/N?: Y

Please wait while your Bitcoin address with custom private key HEX is being created...

-----------------------------------------------------------------------
0: Private Key: 2e79357628ef251e7b7d6c73472fb698cc651881bd6221bb8709699eecd723bd
1: Public Key: 037b6d47cd94edac0bc43f975f0506f50c5245950f157d1323d20ed76eaf7e938e
2: SHA-256 public key: 0495563709c0a17e2fa964f7aa346888b165ade5b5c9a78ef93281ddbbb52716
3: RIPEMD-160 hashing on the result of SHA-256 public key: e2c8defa9809e8b0f2faa67b1ce2ecf51b28d250
4: RIPEMD-160 hash with version byte: 00e2c8defa9809e8b0f2faa67b1ce2ecf51b28d250
5-6: SHA-256 hash for RIPEM-160 hash with version byte: a014cfc4fcdebd0cbee193d6e828202a23543b665b74b6e151efb8acf4ff1216

7: CheckSum: a014cfc4
8: 25 Byte Binary Bitcoin Address: 00e2c8defa9809e8b0f2faa67b1ce2ecf51b28d250a014cfc4
9: Bitcoin Address: 1Mg8M7jx4i2gME6LYukyAcr4fnPQdGiZfH

See the address on the Bitcoin blockchain: https://blockchain.com/btc/address/1Mg8M7jx4i2gME6LYukyAcr4fnPQdGiZfH
-----------------------------------------------------------------------------

Bitcoin address has been successfully created.

C:\Users\Andrew>_
```

The most important output is the Private Key (especially if you chose the option to have the program make a random one for you). This is the most important Key to keep private and secure. It also is the most important to save because with this tool documented here (and online at RFCTools.com) you can automatically generate the address and public key.



The CryptoKeysTool output is modeled off of the website RFCtools.com Not sure this website will be up when this is read, but it is a really cool website that I used for testing and validation of this java tool. The Bitcoin and Ethereum address tools on this website basically do the same thing as the CryptoKeysTool.jar. It's a nice backup but certainly less secure as we are visible on the internet. I would use only for testing (if you feel the need) or only use as a last resort if for some reason you can't get the CryptoKeysTool.jar working offline.

You can see below that we get the same results with both tools.

Bitcoin address test tool

This tool will help you to understand technically how a bitcoin address is generated.

(note: 0 – 8 values are hexstring, 9 is string)

0- Private ECDSA Key (aka Bitcoin private key):

2E79357628EF251E7B7D6C73472FB698CC651881BD6221BB8709699EECD723BD

(any random 256-bit number from 0x1 to 0xFFFF FFFF FFFF FFFF FFFF FFFF FFFE BAAE DCE6 AF48 A03B BFD2 5E8C D036 4140)

more info: Secp256k1

Rand Reset

Public key type:

- (●) **Compressed**
- ( ) **Uncompressed**

☑ **Auto next steps**

calculate public key

1- Public ECDSA Key (aka Bitcoin public key):

037B6D47CD94EDAC0BC43F975F0506F50C5245950F157D1323D20ED76EAF7E938E

(copmpressed public keys must be 33byte and start with 02 or 03)

(uncopmpressed public keys must be 65byte and start with 04)

sha-256(public key)

2- SHA-256 hash of 1:

0495563709C0A17E2FA964F7AA346888B165ADE5B5C9A78EF93281DDBBB52716

ripemd160(this)

3- RIPEMD-160 hash of 2:

E2C8DEFA9809E8B0F2FAA67B1CE2ECF51B28D250

add network byte

4- Adding network byte to 3 (bitcoin mainnet is 00):

00E2C8DEFA9809E8B0F2FAA67B1CE2ECF51B28D250

calculate checksum

checksum calculation

5- SHA-256 hash of 4:

2B19C9418B47086294AF9002D85532BCBE6B93EEDF1578AC24D3B05A4AC09DB7

6- SHA-256 hash of 5:

A014CFC4FCDEBD0CBEE193D6E828202A23543B665B74B6E151EFB8ACF4FF1216

7- First four bytes of 6 (this is checksum):

A014CFC4

8- Adding 7 at the end of 4:

00E2C8DEFA9809E8B0F2FAA67B1CE2ECF51B28D250A014CFC4

base58 encode(this)

9- Base58 encoding of 8 (aka bitcoin address):

1Mg8M7jx4i2gME6LYukyAcr4fnPQdGiZfH