



Abstract

Virtual currencies, such as Bitcoin, have recently dominated news headlines. The term virtual currency has been around since 2012. However, virtual currencies as we know them have existed before then. Most virtual currencies can be used to buy goods anonymously, which has made them popular among privacy concerned individuals. Bitcoin's recent spike in popularity has caused a proliferation and dissemination of information regarding its use and advantages to mainstream audiences. Adoption of virtual currencies by the public has increased, which has introduced a novel issue in digital forensic examinations.

This summer, MultiBit, Litecoin, and Darkcoin desktop wallet software was examined for valuable artifacts on Windows 7 and Ubuntu 14.04 operating systems. This was accomplished in a three-fold fashion: Hard drive, Memory, and Network evidence. In all cases, some form of application settings, user timeline information, and transaction logs were among the type of information able to be recovered. The MultiBit wallet contained the greatest amount of user timeline information including a file created upon uninstalling the application. Overall, the three wallets shared remarkable similarity in artifacts generated.

Introduction

Bitcoin's creation is credited to the alias Satoshi Nakamoto, whose identity is still unknown. The idea behind Bitcoin was initially proposed in October 2008 as a "purely peerto-peer version of electronic cash" that would cut out the middle man, i.e. financial institutions.

In the trust model, a trusted third party verifies that the money has not already been spent before allowing a transaction. This obstacle led to Bitcoin's greatest innovation, the blockchain. The blockchain is essentially a public ledger that is made up of blocks containing all previous transactions of the currency (Fig. 1). This blockchain prevents double spending by the verification process each transaction must undergo. This process is designed to take ten minutes to complete and is composed of these basic steps:

- 1. Assign the transaction to a block that is in a queue to be verified
- 2. Confirm coins were signed with the sender's address private key
- 3. Confirm coins have not already been spent by checking the blockchain
- 4. Repeat for all transactions in the block
- 5. Calculate a difficult SHA256 hash of the block plus "nonce"
- 6. Add the block to the blockchain

Materials & Methods

Materials used for this research project include:

One external hard drive

2. VM Workstation and ISO files for operating systems analyzed

3. Bitcoins, Litecoins, and Darkcoins for trading purposes

4. Access to forensic software programs

Cryptocurrency Artifact Analysis

Preston Miller, B.A.¹, Lance Nudd, J.D.², Chris Vance, B.S.¹, Terry Fenger, Ph.D.¹ ¹Marshall University Forensic Science Center, 1401 Forensic Science Dr, Huntington, WV 25701 ²Stroz Friedberg, 32 Avenue of Americas, New York, NY 10013

Results & Discussion



Figure 1. The blockchain is made up of blocks. Each block is made up of hundreds of transactions. The start of the chain, the Genesis block, is blue. The main chain is black and forks in the chain are purple, referred to as "orphan" blocks.

Disk Forensics

Aultibit_2014-08-01 - Notepad

L5:05:53.906 [AWT-EventQueue-0] DEBUG org.multib L5:05:53.906 [AWT-EventQueue-0] DEBUG org.multib L5:05:53.906 [AWT-EventQueue-0] DEBUG org.multib L5:05:53.906 [AWT-EventQueue-0] DEBUG org.multib

:19:11.161 [NioClientManager] INFO com.google :19:12.081 [NioClientManager] DEBUG com.google [30440220421d7f587a9f8d6e7a8cc286a39bf1d5dd

utpoint:9ad3169acc99e22e2f8f9f26d5b349e1b9869 DUP HASH160 [c9a074d45ac907c1d861d38556026 DUP HASH160 [fde6c54b05f81ab98e8f808bf2c60

Figure 2. Demonstrates the process through which an address is generated. This process is primarily only one way. To protect the address used in this figure, the private and public key values were made up.

			ာBookmarks 🔍 Search Hits မဒ္ဒီRecords		Name	Description	File Type	File Ext	File Created	Entry Modified
	🔁 Hom	ne 💵 File Exten	nts 🔒 Permissions 👸 References 📝 H	ash 🕩 🔽 1	2 🗋 InstallRemover7549935916119	. File, Archive, Not Indexed	Log	log	07/17/14 08:53:09AM	07/17/14 08:53:11AM
ibit.network.MultiBitService - MultiBitService#sendCoins - Sent coins has completed ibit.network.MultiBitService - MultiBitService#sendCoins - Sent coins. Transaction hash is 9ad3169a			🔁 Temp	 1 	3 🛱 izpack7771606733487857202.d	Folder, Deleted, Overwritten		d	07/17/14 08:52:27AM	07/17/14 08:53:10AM
ibit.network.MultiBitService - MultiBitService#sendcoins - Sent coins. Transaction hash is 9ad3169a ibit.network.MultiBitService - Added bitcoinController 20511156 as listener to tx = 9ad3169acc99e22 ibit.file.FileHandler - Start of secureDelete			🦢 Temporary Internet Files 🛅 VirtualStore	1	4 izpack7771606733487857202.log	File, Archive, Not Indexed	Log	log	07/17/14 08:52:27AM	07/17/14 08:53:10AM
Transaction Hash					m					
	E Text 🖁	🔁 Hex 🛛 🖓 Doc	: 🐏 Transcript 🔝 Picture 📃 Report 🛽	Console 🔞	Details 🗐 Output 🗌 Lock 🗌 Code	page 🗌 0/544697				
le.bitcoin.core.MemoryPool - [83.40.36.156]:8333: Peer announced new transaction [1] 4146a0b6fdebb2 le.bitcoin.core.Wallet - Saw relevant pending transaction 4146a0b6fdebb2235d05de99e5de8ff7dfb6c14		-07-17T08:52	2:27.514 Phase 1: JarFile: C:\	Program Fil						Phase 1: Extract
dced3a3b5b2bb1ea0bff56bae76dcffa02204cec588a4e3c22caf6ce15abdf0f7c5aa875a62b9f98b05f6850f5f9e062b5e 35723e1d762dc3ceefe7696788c:0 e272ceb2b5e0] EQUALVERIFY CHECKSIG 0.0095 BTC COULT Addresses and Values Deb6791f4782] EQUALVERIFY CHECKSIG 0.0004 BTC COULT Addresses and Values	0300iles 0450.mod. 0600er.ja 0750pack.	(x86)\Java\ .base=C:\Use ar -Dself.m .uninstaller	to C:\Users\\AppData\ \jre7\bin\java.exe -Xmx64m - ers\\AppData\Local\Te mod.class=com.izforge.izpack.u r.SelfModifier 2014-07-17T08:	X:MaxPermS mp\izpack77 minstaller 52:27.857 H	Size=16m -classpath C:\Use 771606733487857202 -Dself.m .Uninstaller -Dself.mod.met Phase 1: Exit 2014-07-17708	AppData\L mod.jar=C:\Program File hod=uninstall -Dself. :52:29.089 Phase 2: Sp	ocal\Te s (x86) mod.pha awning p	mp\izp \ <mark>Multi</mark> se=2 phase	pack7771606733487 iBit-0.5.18\Unins -Dself.memory=64 3: C:\Program	857202.d -Dself taller\uninstall com.izforge.iz Files (x86)\Java
dced3a3b5b2bb1ea0bff56bae76dcffa02204cec588a4e3c22caf6ce15abdf0f7c5aa875a62b9f98b05f6850f5f9e062b5e 35723e1d762dc3ceefe7696788c:0	0300iles 0450.mod. 0600er.ja 0750pack. 0900\jre7 1050ers\ 1200mod.c	(x86)\Java\ .base=C:\Use ar -Dself.m .uninstaller 7\bin\java.e \Ap class=com.iz	<pre>\jre7\bin\java.exe -Xmx64m - ers\</pre>	XX:MaxPermS mp\izpack77 hinstaller 52:27.857 H 6m -class 57334878572 staller -H	Size=16m -classpath C:\Use 771606733487857202 -Dself.m Uninstaller -Dself.mod.met Phase 1: Exit 2014-07-17T08 spath C:\Users\\Ap 202 -Dself.mod.jar=C:\Progr Dself.mod.method=uninstall	AppData\L hod_jar=C:\Program File hod=uninstall -Dself. :52:29.089 Phase 2: Sp pData\Local\Temp\izpac am Files (x86)\MultiBi -Dself.mod.phase=3 -D	ocal\Te s (x86) mod.pha awning ; k777160 t-0.5.1 self.me	mp\izp se=2 phase 673348 8\Unir mory=6	pack7771606733487 iBit-0.5.18\Unins -Dself.memory=64 3: C:\Program 87857202.d -Dsel: nstaller\uninstal 64 com.izforge.i:	857202.d -Dself taller\uninstall com.izforge.iz Files (x86)\Java f.mod.base=C:\Us ler.jar -Dself. zpack.uninstalle

27:19.785 [AWT-EventQueue-0] INFO com.google 27:19.785 [AWT-EventQueue-0] INFO com.google 27:19.785 [AWT-EventQueue-0] INFO com.google 27:19.785 [AWT-EventQueue-0] INFO com.google cc99e22e2f8f9f26d5b349e1b986985723e1d762dc3ceefe7696788c:1 hash160:49d5d227d6143d21def6ca5e165df97f3cedffb0 Input Address
[1e64eb9041df47cb0186b31e5b6be1c825447cb9] EQUALVERIFY CHECKSIG 0.01 BTC
[c9a074d45ac907c1d861d3855602e272ceb2b5e0] EQUALVERIFY CHECKSIG 0.0134103 BTC -EventQueue-0] DEBUG o.m.v.s.a.SendBitcoinConfirmAction - The fee after completing the transaction was 1000(Queue-0] DEBUG o.m.v.s.action.SendBitcoinNowAction - Sending from wallet C:\Users\Stroz LLC\AppData\Roaming\MultiBit\mult acc99e22e2f8f9f26d5b349e1b986985723e1d762dc3ceefe7696788c:1 hash160:49d5d227d6 [1e64eb9041df47cb0186b31e5b6be1c825447cb9] EQUALVERIFY CHECKSIG 0.01 BTC [c9a074d45ac907c1d861d3855602e272ceb2b5e0] EQUALVERIFY CHECKSIG 0.0134103 BTC 27:25.401 [AWT-EventQueue-0] DEBUG org.multibit.network.MultiBitService - Ping: [98.186.167.15]:8333 27:25.526 [AWT-EventQueue-0] DEBUG org.multibit.network.MultiBitService - MultiBitService#sendCoins fust about to send coins 27:25.900 [AWT-EventQueue-0] INFO com.google.bitcoin.core.Wallet - commitTx of 2456f14833634389d12380be0215ba55002ac18f67cf88c154ba6b7f46a688 27:25.900 [AWT-EventQueue-0] INFO com.google.bitcoin.core.Wallet - marked 9ad3169acc99e22e2f8f9f26d5b349e1b986985723e1d762dc3ceefe7696788c; 27:25.900 [AWT-EventQueue-0] INFO com.google.bitcoin.core.Wallet - 9ad3169acc99e22e2f8f9f26d5b349e1b986985723e1d762dc3ceefe7696788c prevtx 27:25.900 [AWT-EventQueue-0] INFO com.google.bitcoin.core.Wallet - >pending: 2456f14833634389d12380be0215ba55002ac18f67cf88c154ba6b7f46a6883

Figure 3. Transactions sent from the wallet appear after a "SendBitcoinConfirmAction" entry. Transactions to the wallet appear after a "Peer announced new transaction" entry. These transactions contain the time, transaction hash, address inputs and outputs, and amount traded. This can also be compared with the start and stop times to determine transactions in particular sessions of the application.



Figure 6. A more detailed version of Fig. 2 starting with the generated ESDCA public key, which is made up of its (x, y) position on the curve. The Script value from Wireshark is already the 20-byte product from the RIPEM160 and SHA256 hash of the public key. After removing the first 3-bytes and last 2-bytes from the Wireshark Script value the remaining 20-bytes must be processed. First a checksum must be calculated by sequential SHA-256 hashing of the 20-byte value with a prepended 00. The first 4 bytes of this value is appended to the original 20-bytes and base58 encoded. Then a Network ID must be prepended to arrive at the actual address. For Bitcoin, the Network ID is often 1 or 3. It is recommended that a script is created to automate this process.



Figure 5. The breakdown of the contents of a Litecoin tx message. This process can be repeated for any cryptocurrency using the Bitcoin protocol. Please note that the length of the "Script Signature" (10) is variable and the "Output Count" (12) will determine how many "Transaction Output" fields there are. Additionally, the "Transaction Output Script" value must be processed to obtain the address in a similar fashion described in the Bitcoin section (note the Network ID for Litecoin is typically

Byte offset 54-57 54-57 58-69 70-73 74-77 78 79-82

Figure 7. The packet details of the three main messages in the Bitcoin protocol are detailed. The fields within Transaction Input and Output are not shown in this figure. Note that these packet details have the same structure as Litecoin network traffic or any other cryptocurrency using the Bitcoin protocol.



Figure 4. The izpack log appears to be generated when the user uninstalls the MultiBit program from their system.

Network Forensics

recievetradefro	omUbuntu.pcap [W	ireshark 1.10.0 (S	VN Rev 49790 from /trunk-	1.10)]			_ 🗆 🗙
ile <u>E</u> dit <u>V</u> iew	<u>G</u> o <u>C</u> apture <u>A</u>	nalyze <u>S</u> tatistics	; Telephony <u>T</u> ools <u>I</u> nte	~			
) () 🦼 🔳		K 🔁 Q, 🤇	≥ 🛸 🗳 🖗 🖉 📔		1 🛛 🖉 🔀 🎋 🛛 💢		
ilter: tcp.port =	= 9333		_	Expression Clear	Apply Save		
. Time	So 3:33.941885 10	urce	Destination 7 192.168.139.1	Protocol 144 TCP	Length Info 304 9333 > 36235 [PSH, ACK]	Seq=117 Ack=62 Win=64240 Len=25	50
	3:33.957882 19					Seq=1 Ack=1 Win=64952 Len=55	
			m				÷.
			oits), 304 bytes ca				
					f:b6:03 (00:0c:29:df:b6:03) st: 192.168.139.144 (192.168.139.	144)	
					5 (36235), Seq: 117, Ack: 62, Len		
Data (250				0			
Data: fb [Length:		000000000000000000000000000000000000000	000000e20000005ca6a	0ec			
	29 df b6 03 0	05056f6	cf 90 08 00 45 00)P V	Alternating color scheme to	differentiate items listed	
10 01 22	c0 d0 00 00 8	006 1a a 5	6d e4 a4 43 c0 a8	. "m	.c	and of the new states.	
30 <u>fa f</u> 0	11 7b 00 00 f	bc0 b6 db	17 60 37 ea 50 18 74 78 00 00 00 00	\$u@. ty.`7	1. Packet Magic	12. Output Count	
	00 00 00 00 e 01 80 95 85 6		5c a6 a0 ec 01 00 28 83 40 9e 46 8b	br .#(.@	A.F. Z. Command Name		
	27 24 ca 98 1	4 f8 8d ec	a8 5d 80 a6 2e ed	'\$].	3. Payload Length	Transaction Output	
80 a7 c9	6c 47 a7 15 f	b a6 7a ec	26 57 73 7d 5d 96	k HOE. lg z.&ws		13. Value	
90 09 54 a0 d5 54			47 bf 94 02 21 00 58 e7 f6 1b e7 ba	.TyG .T a.X		14. Script Length 15. Script	
	03 2d b7 96 6 03 52 0a 77 e	1 e4 e4 08	83 aa 64 89 97 b8 80 8e c0 35 a7 69	a .!.R.W	····	lo. ochpt	
d0 d5 d8	07 2c 0d 5c 7	9 Oe e3 62	d3 c8 3e c0 b8 94	,.\yb>	Transaction Input	Transaction Output	
	a0 ff ff ff f a9 14 ad ab e		1a 0d 00 00 00 00 43 46 86 36 e6 49	.T yz .v	7 Dravieve Output Uash	16. Value	
00 bc 5d	b7 24 38 50 c	882 88 ac	92 79 b6 30 00 00 90 02 fb 12 51 68	.].\$8Py.	0 8. Previous Output - Index	17. Script Length	
			88 ac 00 00 00 00	Mwv~	9. Script Length	18. Script	
					10. Script Signature 11. Sequence	19. Block Lock Time or Block ID	
					L. Sequence		

	inv		getdata			tx		
e et	Item	Byte offset	- nem		Byte offset	Item		
3	Packet Structure	0-53	Packet Structure	1	0-53	Packet Structure		
	Data	54-	Data	1	54-	Data		
7	Packet Magic	54-57	Packet Magic	1	54-57	Packet Magic		
9	Command Name (69:6e:76)	58-69	Command Name	1	58-69	Command Name (74:78)		
3	Payload Length		(67:65:74:64:61:74:61)	4.	70-73	Payload Length		
7	Payload checksum	70-73	Payload Length		74.77			
	Count	74-77	Payload checksum		74-77	Payload checksum		
2	Type (01 - TX, 02 - Block, 03 -	78	Count		78-81	Transaction Version		
-	Uknown)	79-82	Type (01 - TX, 02 - Block, 03 - Uknown) Data hash		82	Input Count		
14	Data hash				83-262*	Transaction Input *Variable length - 180 bytes in this example		
		83-114			263	Output Count		
					264-297	Transaction Output *Amount of tx output fields dependent on output coun		
					297-301	Block lock time or Block ID		





Conclusions

Examiners must be aware of common artifacts and the extent of information that is obtainable from evidence. In this study, the best method for observing transactions were network captures in combination with a blockchain lookup utility. However, it is important to find remnants of those transactions on a user's computer. When examining a wallet application, it is vital to determine if there is a corresponding log file because they contain the most user activity and transaction information, outside of a network capture. The cryptocurrency wallet software examined shared remarkable similarity in the amount of information stored on the user's machine for both Windows 7 and Ubuntu 14.04. Network traffic was also similar across the cryptocurrencies studied. The process of parsing Bitcoin's network traffic, such as a tx message, was identical in Litecoin and Darkcoin. Consumers and investors have demonstrated high interest in Bitcoin and virtual currencies as a whole and examiners should be prepared to analyze cryptocurrencies effectively as the likelihood of cases involving them continues to increase.

References

Barber, Simon, et al. "Bitter to Better - How to Make Bitcoin a Better Currency." Financial Cryptography and Data Security (2012).

Duffield, Evan and Kyle Hagan. "Darkcoin: Peer-to-Peer Crypto-Currency with Anonymous Blockchain Transactions and an Improved Proof-of-Work System." (2014).

Katz, J and Y Lindell. Introduction to Modern Cryptography. CRC Press, 2007.

Litecoin Wiki. Comparison between Litecoin and Bitcoin. 22 January 2014. < https://litecoin.info/User:Iddo/ Comparison_between_Litecoin_and_Bitcoin>.

Nakamoto, Satoshi. Bitcoin: A Peer-to-Peer Electronic Cash System. 31 October 2008. < http://nakamotoinstitute.org/ bitcoin/>

Percival, Colin. "Stronger Key Derivation Via Sequential Memory-Hard Functions.: (2009).

Perry, David. Bitcoin Mining in Plain English. 6 September 2012. <http://codinginmysleep.com/bitcoin-mining-inplain-english/>.

Shirriff, Ken. Bitcoins the hard way: Using the raw Bitcoin protocol. < http://www.righto.com/2014/02/bitcoins- hardway-using-raw-bitcoin.html>.

Southurst, Jon. Blockchain's SharedCoin Users Can Be Identified, Says Security Expert. 10 June 2014. < http:// www.coindesk.com/blockchains-sharedcoin-users-canidentified-says-security-expert/>.

Acknowledgments

- Dr. Terry Fenger, MUFSC
- Lance Nudd, Stroz Friedberg
- Michael Younger, Stroz Friedberg
- John Shumway, Stroz Friedberg
- Dan Blank, Stroz Friedberg
- Chris Vance, MUFSC
- Dr. Pamela Staton, MUFSC
- Stroz Friedberg
- Marshall University