Analysis of Cryptocurrency Desktop Wallet Software

Preston Miller

Marshall University

Summer '14

Introduction

Often synonymous with providing anonymity for the acquisition of nefarious goods online, crytptocurrencies, like Bitcoin, have been brought to the forefront after gaining traction due to recent media attention. Increased exposure has publicized the utility of crytptocurrencies and spurred the production of new currencies, like Litecoin and Darkcoin. As more consumers turn to online markets, such as Silk Road, for the purchase of both legal and illegal goods, understanding the channels through which crytptocurrencies travel will be vital in future casework. In anticipation of this, Bitcoin, Litecoin and Darkcoin desktop wallet software was examined for forensic artifacts including software GUID, transaction logs, wallet addresses, access times and user and account settings. To ensure a holistic evaluation the examination was conducted on multiple operating systems in three states: fresh installation, intermediate use and deleted.

Basics of Cryptocurrency Design

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 peer-to-peer version of electronic cash" that would cut out the middle man, i.e. financial institutes¹. Satoshi's idea would

¹ Nakamoto, Satoshi. <u>Bitcoin: A Peer-to-Peer Electronic Cash System</u>. 31 October 2008. http://nakamotoinstitute.org/bitcoin/>.

circumvent the financial institution's trust based model in favor of a cryptographic proof of work model. The benefits of this model are twofold:

- Non-reversible transactions protecting sellers from fraud
- Agreement of address escrow mechanism to protect buyers

One issue that needed to be dealt with was double spending. 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 blockchain². The blockchain is essentially a public ledger that is made up of blocks containing all previous transactions of the currency (Fig. 1).

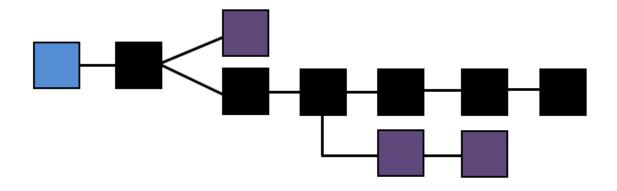


Figure 1. The blockchain is made up of blocks. Each block is made up of hundreds of transactions. The start of the chain, the Genesis block, is blue. The main chain is black and forks in the chain are purple, referred to as "orphan" blocks.

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

² Bitcoin Wiki. <u>Block chain</u>. 21 April 2014. <https://en.bitcoin.it/wiki/Block_chain>.

- 5. Calculate a difficult SHA256 hash of the block plus "nonce"
- 6. Add the block to the blockchain

All the transactions in a given block are considered verified once that block is followed by five blocks. If a block never reaches five following blocks, it is considered an "orphan," as shown in Fig. 1 above.

The majority of the verification process is spent computing an arbitrarily difficult SHA256 hash of the block (step 5, above). Bitcoin uses the hashcash proof of work function for this step³. A SHA256 hash of the block must be calculated with a certain number of preceding zeroes. As more nodes are present (i.e., more Bitcoin users) the number of preceding zeroes increases, effectively increasing the difficulty of the calculation. After every 2,016 verified blocks a system automatically adjusts the difficulty of the hash if the average time of the verification process deviates from ten minutes. Difficulty increases as more Bitcoin nodes are added to the network (Fig. 6). Computers compete by appending a "nonce", a random number, to the block and rehashing it until the correct hash is obtained. Once the correct hash is obtained, the nonce used is broadcasted to the network where each node appends the nonce to the block and hashes it, if a majority of the nodes agree then the block, and all of its transactions, is added to the chain. Once there are five blocks above that block, the transactions are verified. The verification process is referred to as "mining"^{4,5}.

³ Bitcoin Wiki. <u>Hashcash</u>. 8 June 2014. <https://en.bitcoin.it/wiki/Hashcash>.

⁴ Perry, David. <u>Bitcoin Mining in Plain English</u>. 6 September 2012. <http://codinginmysleep.com/bitcoin-mining-inplain-english/>.

⁵ Bitcoin Wiki. <u>Mining</u>. 1 April 2014. < https://en.bitcoin.it/wiki/Mining#The_Computationally-Difficult_Problem>.

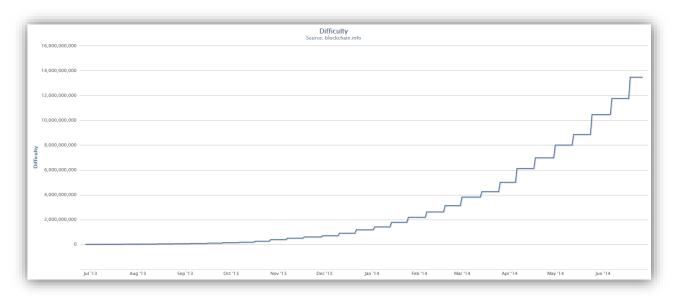


Figure 2. Difficulty is determined every 2,016 blocks and usually increases to prevent the verification process deviating from 10 minutes as more nodes join the network.

Miners are individuals who verify transactions by computing difficult hashes. Miners are incentivized to verify transactions with Bitcoin rewards. More miners increase the integrity of transactions by making it more difficult for a bad actor to have control of a majority of the verification process.

There are three ways of obtaining Bitcoins: purchase from an exchange, transfer among individuals, and mining. Mining has two main functions:

- Verify and add blocks to the blockchain
- Release new Bitcoins to the market

Once the verification processes is completed, the computer that calculated the correct hash is rewarded with freshly 'minted' Bitcoins called the coinbase. The current reward is roughly 25 Bitcoins per block. Users typically pay fees when making transactions. These fees will also be awarded to the miner to incentivize them to include their transactions in the block. These fees will be increasingly important to miners as the coinbase decreases over time. Miners work in pools, as the likelihood of calculating the hash on their own is slim. When a member of a pool calculates the winning hash the

Page 6

reward is split amongst the group based on each individual's contributed computation power. To stimulate scarcity, Bitcoin has a finite amount of coins that can ever be created, 21 million Bitcoins. There are currently over 13 million coins in circulation. As more coins are mined the reward decreases (Table 1).

 Table 1. Depicts diminishing returns on Bitcoins per block as more Bitcoins are in circulation. Each decrease in reward is referred to as a block reward-halving event.

Block	BTC/block	Year	Start BTC	BTC	End BTC	BTC Increase	BTC Supply % of
				Added		(%)	Limit
0	50.00	2009	0	2625000	2625000	Infinite	12.50
52500	50.00	2010	262500	2625000	5250000	100.00	25.00
105000	50.00	2011	5250000	2625000	7875000	50.00	37.50
157500	50.00	2012	7875000	2625000	10500000	33.33	50.00
210000	25.00	2013	10500000	1312500	11812500	12.50	56.25
262500	25.00	2014	11812500	1312500	13125000	11.11	62.50
315000	25.00	2015	1312500	1312500	14437500	10.00	68.75

Users who possess a full Bitcoin client act as nodes on the Bitcoin network ⁶. Each node possesses its own up-to-date version of the blockchain. Each new block added to the chain contains the hash of the header of the block below it, as well as newly verified transactions, when a majority of the nodes verify the transactions and a set hash is computed (Fig. 3-5). This chain of blocks has one path leading to the first block created, the Genesis block. Forks in the chain are possible if two blocks are created within seconds of each other. Because new blocks always have a hash of the block below it,

⁶ Barber, Simon, et al. "Bitter to Better - How to Make Bitcoin a Better Currency." <u>Financial Cryptography and Data</u> <u>Security</u> (2012).

forks never merge further up the chain. Only blocks in the long chain will continue to grow. The unused block is referred to as an Orphan block. The transactions in the Orphan block are added to another block and go through the verification process again to get added to the long chain. Additionally, miners receive no reward for Orphan blocks. Transactions will not be verified until five other blocks follow a block. This way, modification of the blockchain is impractical due to the need to regenerate the hash of all the blocks above it.

Height	Age	Transactions	Total Sent	Relayed By	Size (kB)
307973	8 minutes	1553	11,879.93 BTC	Discus Fish	764.11
307972	42 minutes	327	2,615.49 BTC	BTC Guild	192.24
307971	49 minutes	589	2,960.05 BTC	GHash.IO	324.36
307970	57 minutes	946	7,214.04 BTC	BTC Guild	453.07
307969	1 hour 16 minutes	816	6,343.98 BTC	BTC Guild	440.46
307968	1 hour 30 minutes	262	1,167.27 BTC	Discus Fish	128.27

Figure 3. There are numerous websites that allow individuals to view the blockchain, such as blockchain.info. Each block in the blockchain is assigned a number in the chain and is verified along with the transactions within it.

One concern with the blockchain is that if 51% of the nodes are ever controlled by one or one group of individuals the system breaks. Only a majority of nodes are required to agree to verify a transaction. With 51% of the nodes, they can push through or deny any transaction. However, this inherent flaw is only a concern with large mining pools as an individual is unlikely to have the resources to possess 51% of the nodes at a given time.

a10efc5e048da5d3ef6d7e0d2fd1e8ffdb09e4667f8183ee86bc6729c195da0		2014-06-26 20:15
lo Inputs (Newly Generated Coins)	1KFHE7w8BhaEN (F2Pool Mining Address @)	25.24677656 E
	,	25.24677656 B
7e60b82ec09a16a3115cbd4d57a4fc83190cb9d3071d6c5648419588df18451		2014-06-26 19:52
HwRCRaBVeiEC4S2wq34P1g1voUiGQu5KR	1Por6uz4iP294h9Dw3jQ28VD5JP1QkckZ5 1HwRCRa8VeiEC4S2wq34P1q1voUiGQu5KR	1.7 E 9 81378235 F
		11.51378235 B
ab5d13931b6338298f699b2f5e419d464a3c916b7bb2eb70671f37010345aa0		2014-06-26 19:49
2ssZJSYwCsVBe2dv8pN9fa25Tn7m5jtFz	1HSVYxCNVBkU6hgqvh7emgmdCbdpvCyk92 12ssZJSYwCsVDe2dv8pN9fa25Tn7m5itFz	1 E 2 63772873 F
		3.63772873 E
5701d547366f8181848056d97c8051a91b28d5b4ada9c65e184353b278fcd00		2014-06-26 20:15
HxKTyu6DKJF1ePe3UfSvwCgaN7Ep6QTwx	1NBcTKC2673oq4Bep6uEchaPDnUYNyxKfF	0.10048988 E 3.11819596 E
HxkTyu6DKJF1ePe3UfSwcCgaN7Ep6QTwx	1NBcTKC2673oq4Bep6uEchaPDnUYNyxK/F 1HtxKTyu6DKJF1ePe3U/SwcGaN7Ep6QTwx	

Figure 4. Each transaction in a block can be examined to see the addresses and amount of Bitcoins involved in each transaction.

Transaction View information about a bitcoin transaction								
23972b7f1672bf30b326ff992c6a12	26e9cce273f0296718203ff665d823c4db6							
No Inputs (Newly Generated C	oins)	1CjPR7Z5ZSyWk6 (ghash.io @) - (Unspent) 25.08264529 BTC 2 Confirmations 25 08264529 BTC						
Summary		Network Propagation (Click to view)						
Size	157 (bytes)	Kingdom Hamburg Vilnuso Migak						
Received Time	2014-06-26 20:51:38	Ireland Netherland Berlin Belarus						
Reward From Block	307977	London Belgium Germany Prague						
Scripts	Hide scripts & coinbase	Paris Municho Slovakia U						
Relayed by IP 9	176.9.227.249 (whois)	Google Bay of France Zurich Vaduz Zagreb Map data @2014. Soogle						
Visualize	View Tree Chart							

Figure 5. Each individual transaction in a block can be examined for even more data including "Received Time".

Bitcoin for Individual Users

Bitcoin is categorized as a cryptocurrency because of its cryptographic roots^{7,8}. Bitcoins are stored in web, mobile, or desktop wallets and, within a wallet, are assigned to certain user created addresses. For each address, a public and private key is generated. Users can dictate which address to use for any transaction. This is confirmed in the verification process with the user's public key that is mathematically related to the private key.

Bitcoins wallet addresses and public and private keys are mathematically linked through a series of algorithms (Fig. 6). This process is described in detail in Ken Shirriff's blog⁹. Private keys are randomly generated 32-byte numbers. Bitcoin uses Elliptic Curve Digital Signature Algorithm (ESDCA) cryptography to generate a 64-byte public key with a 0x04 prefix from the private key. The SHA-256 and RIPEM 160 algorithms create a 20-byte hash of the public key. The address associated with the public and private key is simply a 1 byte prefix, the custom base58 encoding of the 20-byte public key hash, and a 4 byte check code. This process can be replicated forward starting with a random private key to arrive at an address. However, it has not been proven possible to use an address and derive its private key. In a transaction, the sender signs the coins with the used address' private key. The blockchain then confirms that the coins being used have not already been spent, through the verification process.

⁷ Katz, J and Y Lindell. Introduction to Modern Cryptography. CRC Press, 2007.

⁸ IEEE. <u>Standard Specifications For Public-Key Cryptography</u>. 10 October 2008. http://grouper.ieee.org/groups/1363/.

⁹ Shirriff, Ken. <u>Bitcoins the hard way: Using the raw Bitcoin protocol.</u> < http://www.righto.com/2014/02/bitcoinshard-way-using-raw-bitcoin.html>

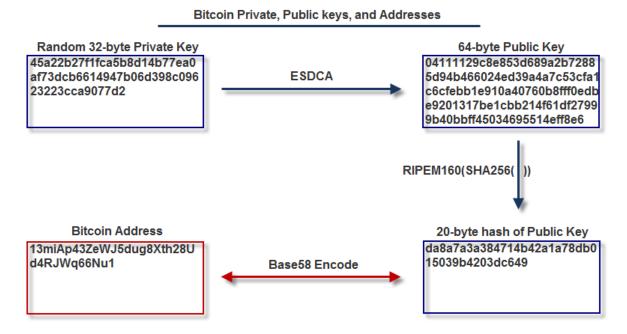


Figure 6. Demonstrates the process through which an address is generated. This process is primarily only one way. To protect the address used in this figure, the private and public key values were made up.

Litecoin

Litecoin is based on Bitcoin's node system and was created by Charles Lee in October 2011¹⁰. The currency was created to make improvements on the Bitcoin system. Overall, the currency works in the same manner as Bitcoin. Litecoin has a peer-to-peer network where each node has an updated blockchain. Litecoins are stored in wallet addresses with public and private keys. The sender's private key is used to sign the currency when making a transaction. Verification proceeds with miners packaging transactions in blocks, confirming a sender's legitimate private key was used to sign the currency by comparing it to the sender's public key, and processing a difficult hash. Miners that calculate the correct hash are rewarded with Litecoins. Litecoin has four main variations on the Bitcoin model¹¹:

• Scrypt based hash

¹⁰ Litecoin Wiki. Litecoin. 20 January 2014. <https://litecoin.info/Litecoin>.

¹¹ Litecoin Wiki. <u>Comparison between Litecoin and Bitcoin</u>. 22 January 2014. https://litecoin.info/User:Iddo/Comparison_between_Litecoin_and_Bitcoin.

- Four times faster verification time
- More frequent difficulty adjustments
- Four times total Litecoin limit

Scrypt is a memory-hard function whose purpose is to maximize the amount of memory it uses for every process¹². Scrypt was originally created to make brute forcing a password hash more time consuming, however it has been implemented in Litecoin to decrease entry cost. Litecoin is more memory intensive rather than just arithmetic based, like Bitcoin. Application-specific integrated circuits, ASICs, are chips that are designed for specific purposes, such as Bitcoin mining. ASICs, which are more commonly used in Bitcoin, would be considerably more expensive for Litecoin mining. Additionally, return benefits from ASIC hardware would also be decreased by 10-fold in Litecoin mining. CPU mining is no longer truly viable with Bitcoin. It is, however, still an option for Litecoin miners. The difference between CPU vs. GPU mining in Litecoin is less extreme than in Bitcoin and because of these factors; Litecoin is more inclusive in that relatively inexpensive hardware can be used to productively mine.

Litecoin boasts a two and a half minute verification time. This also means that miners are rewarded more frequently as blocks are verified more often. Because the 2,016 blocks are processed four times quicker than in Bitcoin, this means that forks in the blockchain and difficulty readjustments happen four times more frequently. As forks happen more frequently, there is a greater waste of computational power as nodes are more likely to hash orphan blocks before the location of the correct block is broadcasted. With the potential for there to be 84 million Litecoins in circulation, the market cap and purchasing power of Litecoin would surpass Bitcoin, if the value of a Litecoin were greater than a quarter of a Bitcoin; however, currently a Litecoin is only worth two percent of a Bitcoin.

Darkcoin

¹² Percival, Colin. "Stronger Key Derivation Via Sequential Memory-Hard Functions." (2009).

Darkcoin is a relative newcomer to the cryptocurrency field, but has grown quickly perhaps owing to its increased focus on anonymity¹³. It does this through the custom implementation of CoinJoin, called Darksend. CoinJoin is a popular structure that many mixing services are based on. While CoinJoin is not built into Bitcoin, Dark wallet is a CoinJoin implementation built on the Bitcoin network. CoinJoin works by mixing multiple transactions into a pool and then sending money from that pool to intermediaries, other pools, and eventually to the recipients. However, this mixing service is provided by a centralized server, which if it were compromised 'anonymous' transactions would be revealed. Another mixing service based on CoinJoin is ShareCoin and was recently discovered to not be as anonymous as users believed¹⁴. Since Darksend is built into the Darkcoin client, their mixing service is decentralized and is unlikely that enough users would be compromised to threaten anonymity on the server. As yet, no weakness has been discovered in Darksend's capability to provide anonymous transactions.

Darkcoin has three unique properties: denominations, master nodes, and an inflationary currency. As of Release Candidate 4, to use Darksend users must allow their wallet to 'anonymize' their Darkcoins first and then only send transactions using the anonymized Darkcoins. The anonymizing process is configurable and shows up in the wallet's transaction log with the type "Darksend Denominate" with an associated fee of 0.001 Darkcoins for each mixing cycle. Master nodes are simply nodes that provide the mixing service using the Darksend technology. These master nodes are elected by a pseudorandom algorithm and receive twenty percent of the block reward. Becoming a master node

¹³ Duffield, Evan and Kyle Hagan. "Darkcoin: Peer-to-Peer Crypto-Currency with Anonymous Blockchain Transactions and an Improved Proof-of-Work System." (2014).

¹⁴ Southurst, Jon. <u>Blockchain's SharedCoin Users Can Be Identified, Says Security Expert</u>. 10 June 2014. http://www.coindesk.com/blockchains-sharedcoin-users-can-identified-says-security-expert/.

is meant to be difficult, a node must have at least 1,000 Darkcoins and the user must be technically savvy enough to set up the master node.

Darkcoin utilizes a two and a half minute verification process, similar to Litecoin. There are only a maximum of 22 million Darkcoins that can be created. Darkcoin uses X11 as its proof of work hash. X11 is a chained hashing algorithm that is composed of 11 different hashes. This hashing function is more complicated than the SHA256 hash in Bitcoin and prevents current ASICs from being utilized efficiently. Darkcoin, like Litecoin, favors CPU and GPU mining. Darkcoin revaluates the difficulty of the hash every block using DarkGravityWave. In contrast to Bitcoin and Litecoin, Darkcoin features a block reward curve where reward decreases gradually as more blocks are mined. The reward curve will eventually reach a steady 7% annual release of Darkcoins. This makes Darkcoin an inflationary currency, unlike the deflationary Bitcoin and Litecoin.

This introduction is not meant to address every nuance of these currencies, but should provide sufficient working knowledge of these cryptocurrencies for this report. A reference table below summarizes the differences between the three cryptocurrencies under examination. (Table 2).

	Bitcoin	Litecoin	Darkcoin
Current value (USD)	600	9	9
Coins in circulation	13,000,000	30,000,000	4,500,000
Currency type	Deflationary	Deflationary	Inflationary
Maximum coins	21,000,000	84,000,000	22,000,000
Verification time (min)	10	2.5	2.5
Hash used	SHA256	Scrypt	X11

Table 2. Summarizes the main differences between the three cryptocurrencies examined in this paper.

Reward	25 BTC	50 LTC	5 DRK
Reward type	Reward halving	Reward halving	Reward curving
Difficulty adjustment	2,016 transactions	2,016 transactions	1 transaction

Materials & Methods

Materials used for this research project include:

- One external hard drive
- VM Workstation and ISO files for operating systems analyzed
- Bitcoins, Litecoins, and Darkcoins for trading purposes
- Access to forensic software programs (listed in Table 3)

Virtual machines (VMs) were created and maintained by VMware Workstation version 9.0.0. Each operating system analyzed had a negative control and three states: no use, intermediate use, delete. Each cryptocurrency wallet had a unique VM to avoid potentially confusing artifacts from a different wallet software. Negative controls were created with fresh installations of the operating systems being analyzed. The control's memory and hard drive were preserved for later analysis by duplicating the VM's VMEM and VMDK files, respectively. Controls were then copied using VMware's "Full Clone" option to create independent VMs for testing.

The first batch of clones, labeled the "Before" state, had their respective wallets installed and up to date with the blockchain, but were not used to trade. Clones of the "Before" state, labeled the "After" state, were used to conduct trades with a variety of other wallets. In order to obtain a comprehensive dataset, wallets sent to and received from included desktop, mobile, and web wallets. Clones of the "After" state, labeled the "Delete" state, employed a variety of deletion methods and the resulting artifacts upon deletion of the software. On Windows VMs, Regshot version 1.8.3 was used to analyze differences in the registry for all VM states. Additionally, Wireshark version 1.6.1 captured network activity for all VM states. Each VM's VMEM file was preserved before shutting it down. Afterwards a copy of their VMDK was also preserved for analysis. VMDK files were converted to an E01 image with FTK Imager Version 3.1.3.2. A variety of tools were used for memory and hard drive analysis (Table 3).

Software	Version	Purpose
Memory Analysis		
FTK Imager	3.1.3.2	String searching
Volatility	2.3.1	Overall memory analysis toolkit
MantaRay	1.2.9	Overall memory analysis toolkit
Hard Drive Analysis		
EnCase	6.17.0.90 & 7.08.00.137	Overall case analysis toolkit
Linkil	1.0.0.1	Lnk file analysis

 Table 3. Tools used, version, and their purpose in analysis.

Contemporaneous notes were maintained throughout the entire research project. Where possible, results were placed in an Excel workbook on separate spreadsheets for convenience. Relevant artifacts searched for include: transactions with date and time stamps, GUID, user settings, installation information, and usage.

The "Deleted" state simulated deletion of cryptocurrency software and their artifacts using multiple methods. The methods used were: uninstall program with the provided uninstaller or CCleaner version 4.11.4619 uninstall and clean modules. The memory and hard drive were preserved and analyzed to identify any new or remaining artifacts resulting from a range of shallow to deep deletion activity.

Results & Discussion

Bitcoin – MultiBit 0.5.18

Multiple transactions were conducted with the MultiBit 0.5.18 client on Windows 7 and Ubuntu 14.04 VMs. After imaging and analysis of the virtual drives, the resulting artifacts were similar on both

operating systems. The majority of pertinent artifacts were found in

C:\Users\[User]\AppData\Roaming\MultiBit and /home/[User]/MultiBit directories on Windows and Ubuntu, respectively. There was also a MultiBit-0.5.18 folder in C:\Program Files(x86) and /home/[User] directories. However the MultiBit-0.5.18 folder did not contain any user activity artifacts. Limited transaction data was stored in multibit .info files, located in the previously mentioned directories (Fig. 7). In my research, these .info files contained only transactions sent from the wallet.

Home 📴 Entries 🛈 Bookmarks 🔍 Search Hits 🥰 Records 🜗		Name	Description	File Type	File Ext	File Created	Entry Modified		
🔁 Home 🕫 File Extents 🔒 Permissions 👸 References 📝 Ha ()	4	D multibit-20140630122444.info	File, Archive, Not Indexed		info	06/30/14 12:24:44PM	06/30/14 12:24:44PM		
P-D 🗋 MultiBit	2 5	D multibit-20140630150122.wallet	File, Archive, Not Indexed		wallet	06/30/14 03:01:22PM	06/30/14 03:01:22PM		
	2 6	D multibit-20140630150122.info	File, Archive, Not Indexed		info	06/30/14 03:01:22PM	06/30/14 03:01:22PM		
😑 🕞 🔂 🧰 multibit-data	7	D multibit-20140630150619.wallet	File, Archive, Not Indexed		wallet	06/30/14 03:06:19PM	06/30/14 03:06:19PM		
-D_ Ca key-backup	28	D multibit-20140630150619.info	File, Archive, Not Indexed		info	06/30/14 03:06:19PM	06/30/14 03:06:19PM		
-D_ C C rolling-backup	2 9	D multibit-20140630153129.wallet	File, Archive, Not Indexed		wallet	06/30/14 03:31:29PM	06/30/14 03:31:29PM		
wallet-backup	 10 	T multihit_20140630153120 infn III	File Archive Not Indexed		info	06/30/14 03:31:30DM	06/30/14 03:31:30DM		
🔚 Text 🐙 Hex 🎧 Doc 🔚 Transcript 🔝 Picture 📃 Report 💽 Cor	nsole 🔞) Details 🗊 Output 🛛 Lock 🔽 Code	epage 🗌 0/547563						
0145 <mark>r4gnJ73yHa8FMj,First%20Trade%20to%20AJ</mark> property,r 0290rty, <mark>wallet</mark> BackupFile,C%3A%5CUsers%5CStroz%20LLC%5	0000_ultiBit.info,1 walletVersion,3 receive,17jQUYRISJSPqPcczZHWcuFYMcCpkUhX1D, receive,IKP73GB6talRLAH5LZqMLd9YkE2yGLZA7P, send,1Q9WQm8DJLsRLycPw5V1 0145r4gnJ73yHa8FMj,First*20Trade*20to*20AJ property, receiveAddress,17jQUYRISJSPqPcczZHWcuFYMcCpkUhX1D property,validationAmountIsTooSmall, false prope 0290rty,walletBackupFile,0*3A*5CUSers*5CStrox*20LLC%5CAppBat*5CRomming*5CMultiBit*5Cmultibit-dat*5Crolling-backup*5Cmultibit-20140630150614,wallet 0435property,walletBescription,Wy*20MultBit*20MultBit*20LLC%5CAppBat*5CRomming*5CMultBit*5Cmultibit-dat*5Crolling-backup*5Cmultibit-20140630150614,wallet								
0880Address,1Q9WQm8DJLsRLycPw5V1r4gnJ73yHa8FMj proper 0725 0870	0580 <mark>Address,1Q9WQm3DJLsRLycPw5V1r4gnJ73yHa8FMj property,sendLabel,First%20Trade%20to%20AJ</mark> 0725								
ono 101500-0-9HECLPE=Conumbt confricther inumber: confrictimens clorging ifactiled, duer storinergootiatii 1660 m. failluire witch starget: confricther FailuRepes_00-00-0-2_00-09 _NAME=Loging in cother: F 1305 airled, Responderes cologingotherFaillRepes_00-00-0-2_00-09 _HELPE=Coundro of cloging in cother: v									
C\Users\\AppData\Roaming\MultiBit	multibit-c	lata\wallet-backup\multibit-20140630150	<mark>0619.info</mark> (PS 287000 LS 284	952 CL	35619	SO 000 FO 0 LE 1)			

Figure 7. The multibit .info files stored in the MultiBit\multibit-data\wallet-backup\ directory contain some transaction data. New .info files appear to be generated as part of MultiBit's wallet backup service.

The MultiBit 0.5.18 and the Uninstall MultiBit 0.5.18 lnk files contained the SID of the account that

had the application installed (Fig. 8). This information is helpful in determining ownership of the

software program.

Home Entries DBookmarks 🔍 Search Hits 🖏 Records 🔸		Description	File Type	File Ext	File Created	Entry Modified		
🔚 Home 🕫 File Extents 🧯 Permissions 🕅 References 📝 Ha 🕩	🖌 🖸 1 🗋 desktop.ini	File, Hidde	Initialization	ini	07/14/09 12:54:23AM	12/12/12 12:00:41PM		
-D 🛛 🦣 Default User 🔹 🔺	2 🥯 MultiBit 0.5.18.lnk	File, Archive	Windows File Shortcut	Ink	06/23/14 02:53:20PM	06/23/14 02:53:20PM		
📮 🖓 🚞 Public								
-D C C Desktop								
🗊 — 🕞 🔂 Documents								
-Dm 🛅 Downloads								
-D 🔁 🚞 Favorites								
-Dm 🛅 Libraries								
🕒 🕞 🔂 Music								
D C C C C C C C C C C C C C C C C C								
PLOCI C Recorded TV								
📴 Text 💐 Hex 🎧 Doc 🐏 Transcript 🔛 Picture 📃 Report 🖸 Col	onsole 🔞 Details 🍺 Output 🗌 Lock 🗌 Co	odepage 🛛 0/	547563					
0000LÀFÝ@								
0145Files (x86) T								
0290 · · · · · · · · · · · · · · · · · · ·								
0580 & C::\\P:r:o:g:r:a:m: F:i:l:e:s: (:x:8:6:):\								
0725051.8.\.m.u.l.t.i.b.i.ti.c.o%Pr	rogramFiles%\MultiBit-0.5.18\mult	ibit.ico						
0870								
1015 · · · · · · · · · · · · · · · · · · ·	u-1-t-1-B-1-t-=-051-B-\-m-u	1.1.2.1.0.1	·t·1·c·o·····					
1305								
1450					£·····i@Z û			
	-M-11-1-+	SELVENT ROWER	• & m Îm • • • • • • • • • • •		· S · = · 1 · = · 5 · = · 2 · 1 ··	2 4 0 0 0 1 1		
15950 ···· A ···15DSå (bù0h · « ···+'·04 ···································								
1740 -4 -5 -23 -3 -1 -0 -7 -6 -3 -1 -7 -83 -6 -3 -0 -4 -0 -0 -8 -1 -								

Figure 8. The MultiBit 0.5.18 and Uninstall MultiBit 0.5.18 lnk files contain the SID of the user's account that the program is installed on.

The multibit.log file found in the MultiBit\log\ folder on both Windows and Ubuntu VMs contains a wealth of information including program start and stop times and detailed transaction logs. Time stamps in the log were primarily in local time, unless otherwise stated. For example, when a user starts MultiBit, a "Starting MultiBit at" followed by a date and time stamp in UTC appears in the log. The last entry added to the log when a user closes MultiBit is "Making request to shut down socket" preceded by a local time stamp (Fig. 9). It is possible for a start entry to not be followed by a corresponding stop entry, and this indicates improper termination of the application such as a forced shutdown of the operating system. Only the most recent transactions will be found in the log as it has a finite storage capacity.

🗍 output - Notepad	
File Edit Format View Help	
START on line: 0 14:53:48.255 [main] INFO org.multibit.MultiBit - Starting MultiBit at 23 Jun 2014 18:53:48 GMT ◀ START on line: 726	Two consecutive start commands - result from improper termination of the application.
	eer begins application 5/30/14 16:24:31 UTC
12:31:15.183 [AwT-EventQueue-O] DEBUG o.m.ApplicationInstanceManager - Making request to shut down socket START on line: 1103	User ends application 06/30/14 16:31:15 UTC
15:01:09.140 [main] INFO org.multibit.MultiBit - Starting MultiBit at 30 Jun 2014 19:01:09 GMT	

Figure 9. Start and stop commands from the multibit.log file can help an examiner establish a user activity timeline.

Detailed transaction data was stored in the multibit.log file. Each transaction sent from the wallet

begins with a "SendBitcoinConfirmAction" preceded by a local timestamp. The entries following contain

the transaction hash, addresses, and the amount of BTC involved. Note, to obtain the public addresses,

one must convert the RIPEM-160/SHA-256 representation of the public address found in the log into

Base58. This process will be demonstrated in the network capture section.

A "Peer announced new transaction" is indicative of a transaction being received by the wallet. Like

the above, a local timestamp, transaction hash, addresses, and the amount of BTC involved are

preserved (Fig. 10). A python script searching for the terms indicating start and stop activity, and

transactions would be ideal for analyzing this lengthy log file.

Multibit 2014-08-01 - Notepad
The Edit Format View Help
15:05:53.906 [AWT-EventQueue-0] DEBUG org.multibit.network.MultiBitService - MultiBitService#sendCoins - Sent coins has completed 15:05:53.906 [AWT-EventQueue-0] DEBUG org.multibit.network.MultiBitService - MultiBitService#sendCoins - Sent coins. Transaction hash is 9ad3169a 15:05:53.906 [AWT-EventQueue-0] DEBUG org.multibit.network.MultiBitService - Added bitcoinController 20511156 as listener to tx = 9ad3169acc99e22 15:05:53.906 [AWT-EventQueue-0] DEBUG org.multibit.file.FileHandler - Start of secureDelete
TRANSACTION Received on line: 1699
15:19:11.161 [NioClientManager] INFO com.google.bitcoin.core.MemoryPool - [83.40.36.156]:8333: Peer announced new transaction 1] 4146a0b6fdebb2 15:19:12.081 [NioClientManager] DEBUG com.google.bitcoin.core.wallet - Saw relevant pending transaction 4146a0b6fdebb2235d05de99e5de8ff7dfb6c14 in [3040220421d7587a9486d6r2a6c286a39b51d5dce3a3b5b2bblea0bff56bae76dcffa02204ce588a4e3c22caf6ce15abdf0f7c5aa875a62b998b5f6850f5f9e062b5e outpoint:9ad3169acc99e22e2f8f9f26d5b349e1b986985723e1d762dc3ceefc766788c:0 out DUP HASH160 [94074d5ac907c1d861d3855602e72ceb2b5e0] EQUALVERIFY CHECKSIG 0.0095 BTC - Output Addresses and Values
15:19:12.081 [NioClientManager] INFO com.google.bitcoin.core.Peer - [83.40.36.156]:8333: Downloading dependencies of 4146a0b6fdebb2235d05de99e5d
TRANSACTION Sent on line: 1773
15:27:19.785 [AWT-EventQueue-0] DEBUG o.m.v.s.a.SendBitcoinConfinmAction - Just about to complete the tx (and calculate the fee) 15:27:19.785 [AWT-EventQueue-0] INFO com.google.bitcoin.core.wallet - Completing send tx with 1 outputs totalling 1000000 satoshis (not includin 15:27:19.785 [AWT-EventQueue-0] INFO com.google.bitcoin.core.wallet - with 0.0134103 coins change 15:27:19.785 [AWT-EventQueue-0] INFO com.google.bitcoin.core.wallet - with 0.0134103 coins change 15:27:19.785 [AWT-EventQueue-0] INFO com.google.bitcoin.core.wallet - with 0.424103 coins change 15:27:19.785 [AWT-EventQueue-0] INFO com.google.bitcoin.core.wallet - completed: 27493764adff13ee641eb055f943542198e0aa5dbc1c421a1c5b22982eb
in outpoint:9ad3169acc99e22e2f8f9f26d5b349e1b986985723e1d762dc3ceefe7696788c:1 hash160:49d5d227d6143d21def6ca5e165df97f3cedffb0
15:27:19.785 [AWT-EventQueue-0] DEBUG o.m.v.s.a.SendBitcoinConfirmAction - The fee after completing the transaction was 10000 15:27:25.401 [AWT-EventQueue-0] DEBUG o.m.v.s.action.SendBitcoinNowAction - Sending from wallet C:\Users\Stroz LLC\AppData\Roaming\MultiBit\multi
in outpoint:9ad3169acc99e22e2f8f9f26d5b349e1b986985723e1d762dc3ceefe7696788c:1 hash160:49d5d227d6143d21def6ca5e165df97f3cedffb0 out DUP HASH160 [1e64eb9041df47cb0186b31e5b6be1c825447cb9] EQUALVERIFY CHECKSIG 0.01 BTC out DUP HASH160 [c9a074d45ac907c1d861d3855602e272ceb2b5e0] EQUALVERIFY CHECKSIG 0.0134103 BTC
15:27:25.401 [AWT-EventQueue-0] DEBUG org.multibit.network.MultiBitService - Ping: [98.186.167.15]:8333 Transaction Hash 15:27:25.526 [AWT-EventQueue-0] DEBUG org.multibit.network.MultiBitService - MultiBitService#sendcoins Just about to send coins 15:27:25.900 [AWT-EventQueue-0] INF0 com.google.bitcoin.core.wallet - commitx of 2456f14833634389d12380be0215ba55002actBf676f88ct154ba6b7f46a688 15:27:25.900 [AWT-EventQueue-0] INF0 com.google.bitcoin.core.wallet - marked 9ad3169acc99e22e2f8f9f26d5b349e1b986985723e1d762dc3ceefe7696788c: 15:27:25.900 [AWT-EventQueue-0] INF0 com.google.bitcoin.core.wallet - marked 9ad3169acc99e22e2f8f9f26d5b349e1b986985723e1d762dc3ceefe7696788c: 15:27:25.900 [AWT-EventQueue-0] INF0 com.google.bitcoin.core.wallet - >pending: 2456f14833634389d12380be0215ba55002actBf67cf88c154ba6b7f46a6883
۲ III ا

Figure 10. 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.

RecentFileCache.bcf in the C:\Windows\AppCompat\Programs\ directory is another source that can

reference recent execution of an application (Fig. 11). Previous research on RecentFileCache.bcf

suggests that listed executables are typically new to the system and recently run by the user¹⁵.

📇 Home 🎏 Entries 💭 Bookmarks 🔍 Search Hits 🕫 Records 📣		Name	Description	File Type	File Ext	File Created	Entry Modified
🔁 Home 📴 File Extents 🔒 Permissions 🛱 References 📝 Ha 🕩	🛛 1	AEINV_PREVIOUS.xml	File, Archive	XML Document	xml	06/17/14 05:33:28PM	06/17/14 05:33:37PM
-D 🗋 Windows 🔺	2	RecentFileCache.bcf	File, System, Archive		bcf	07/14/09 01:08:52AM	06/30/14 04:48:56PM
-D_ in \$Reconfig\$							
-D C C addins							
-D Compat							
LD Programs							
🕀 🕞 🔂 AppPatch							
🕀 🕞 🔂 assembly							
BitLockerDiscoveryVolumeContents	•	III					
💽 Text 🤯 Hex 🍙 Doc 🐏 Transcript 🖾 Picture 📃 Report 💽 Cor	sole 🕅	🕽 Details 🏼 🗐 Output 🔽	Lock 🗌 Codepage 🗌 0/5	547563			
0000 ÿîÿ·"·····B÷é°\$···c·: \ w·i·n·d·o·w·s·\ s·y							
0145 ·e · \ · v ·m ·w ·a ·r ·e · · t ·o ·o · l ·s · \ · v ·m ·w ·a ·r ·e ·r ·e ·s · 0290 e · · ·A · · ·c ·: · \ ·u ·s ·e ·r ·s · \ ·							
0435 ·: · \ · p · r · o · q · r · a · m · · f · i · l · e · s · · (· x · 8 · 6 ·) · \ · j · a ·							
05806·)·\·c·o·m·m·o·n· ·f·i·l·e·s·\·j·a·v·a·\·j·a·v·a							
0725 · \ · j · r · e · 7 · \ · b · i · n · \ · j · p · 2 · l · a · u · n · c · h · e · r · . · e · x ·							
0870\\s.y.s.t.e.m.3.2.\\f.s.q.u.i.r.te.x.e@c							
1015 ·a·t·e·\·j·u·s·c·h·e·d·.·e·x·e···&···c·:·\·p·r·o·							
1160s t e m · 3 · 2 · \ · v · e · r · c · l · s · i · d · . · e · x · e · · · > · · · c · : · \ 1305 <mark>· s s · e · t · u · p · . · e · x · e</mark> · · · / · · · c · : · \ · p · r · o · g · r · a · m ·			·\ ·d ·o ·w ·n ·l ·				
1450w·s·\·s·y·s·t·e·m·3·2·\·i·p·c·o·n·f·i·g·.·e·x·e··							
C\Windows\AppCompat\Programs\RecentFile(ache.b	<mark>cf</mark> (PS 2093112 LS 209106	4 CL 261383 SO 000 FO 0	LE 1)			

Figure 11. RecentFileCache.bcf contains recently executed applications and may contain reference to an executable even if it is no longer present on the system.

There are registry keys present in the SOFTWARE hive related to MultiBit, however no valuable user

information was apparent. The following keys were associated with MultiBit installation:

- HKLM\SOFTWARE\Classes\bitcoin
- HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\MultiBit

0.5.18

After thorough analysis, MultiBit was uninstalled on both Windows and Ubuntu VMs. In Windows,

the only pertinent files removed were lnk files and registry keys. In both cases, the MultiBit directory

containing the .info and multibit.log files were still present. Equally as important, a log file titled izpack

appended with a series of numbers, is created when the user uninstalls MultiBit (Fig. 12). This file

appears in the C:\Users\[User]\AppData\Local\temp and /tmp/ directories on Windows and Ubuntu,

¹⁵ Harrell, Corey. <u>Revealing the RecentFileCache.bcf File</u>. 2 December 2013.<http://journeyintoir.blogspot.com/2013/12/revealing-recentfilecachebcf-file.html>.

respectively. IzPack is a packaging tool for Java applications. MultiBit's universal Java installer is based off of IzPack and is most likely the source of this deletion artifact. No reference to IzPack was found on

the system prior to uninstalling MultiBit.

🕒 Home 🚡 Entries 🗊 Bookmarks 🔍 Search Hits 🭕 Records 🖕 👀		Name	Description	File Type	File Ext	File Created	Entry Modified			
🔁 Home 🕫 File Extents 🔒 Permissions 🛱 References 📝 Hash 🕩	2 12	InstallRemover7549935916119	File, Archive, Not Indexed	Log	log	07/17/14 08:53:09AM	07/17/14 08:53:11AM			
⊕-D_2 🗁 Temp 🔨	🛛 13	C izpack7771606733487857202.d	Folder, Deleted, Overwritten		d	07/17/14 08:52:27AM	07/17/14 08:53:10AM			
	🛛 14	izpack7771606733487857202.log	File, Archive, Not Indexed	Log	log	07/17/14 08:52:27AM	07/17/14 08:53:10AM			
VirtualStore 👻	•	III								
🗈 Text 🐙 Hex 🎧 Doc 🧏 Transcript 🖾 Picture 📄 Report 🖸 Console 🧃 Details 🖉 Output 🔽 Lock 💭 Codepage 💭 0/544697										
00002014-07-17709:52:27.514 Phase 1: JarFile: C:\Progra 0150ed 149 files into C:\Users\AppData\Local\Temp\izp 0300iles (x86)\Java\jre7\bin\java.exe -Xmx64m -XX:Max 0450 mod.base=C:\Users\AppData\Local\Temp\izp 0600er.jarAppData\Local\Temp\izp 05000\jre7\bin\java.exe -Xmx64m -XX:Max 0750pack.uninstaller.SelfModifier 2014-07-17708:52:27. 05000\jre7\bin\java.exe -Xmx64m -XX:MaxPermSize=16m - 1050ers\AppData\Local\Temp\izpack777160673348 1200mod.class=com.izforge.izpack.uninstaller.Uninstaller 15002:30.010 Phase 3: Method returned, waiting for othe 1650s 4: Teturn value = 0 1800 CUsers\AppData\Local\Temp\zpaca	Temp\: PermS: ack77' ller.U 857 P class 785720 r -D voking r thre	<pre>izpack771606733487857202.d ize=16m -clasppath C:\Use ize=16m -clasppath C:\Use ize=16073348785720 -DBelf.mod.meth mase 1: Exit 2014-07-17T08 path C:\Users\Ap 22 -DBelf.mod.methOd=uninstall g method: com.izforge.izpaci</pre>	2014-07-17T08:52:27. Ts\ TapDtat\L doi_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 k.uninstaller.Uninstal 99 Phase 2: deleteing	841 Pha ocal\Te s (x86) mod.pha awning k777160 t-0.5.1 self.me ler.uni sandbox	se 1: mp\iz \ <mark>Mult</mark> se=2 phase 67334 8\Uni mory= .nstal 201	Spawning phase 2 pack7771606733487 iBit-0.5.18\Unins -Dself.memory=64 3: C:\Program i 87857202.d -Dsel: nstaller\uninstal: 64 com.izforge.i: 1(String[] args)	: C:\Program F 557202.d -Dself taller\uninstall com.izforge.iz Files (x86)\Java f.mod.base=C:\Us ler.jar -Dself. zpack.uninstalle 2014-07-17T08:5			

Figure 12. The izpack log appears to be generated upon the user uninstalling the MultiBit program from their system.

These artifacts can be used to determine the following: which account the program is installed

on, recent usage, user activity timeline, date and time of uninstallation of the program, and detailed

transaction logs.

Due to issues with the VM's VMEM files, string searching was the most extensive memory analysis

possible. This revealed wallet addresses and sent/received address references, but did not allow for the

recovery of transaction data (Fig. 13). This information could be used to indicate that MultiBit had been

used during the current session and provide the examiner with additional addresses associated with the

account.

Q AccessData FTK Imager 3.1.3.2	and show the back of the Back	- • X
<u>File V</u> iew <u>M</u> ode <u>H</u> elp		
🛯 🏩 🎕 🎕 🚔 🖾 🖬 🖬 📾 🛥 🚥 🚥	📮 🔍 🗋 🗎 🖮 🐱 📽 💡 🖕	
Evidence Tree ×	C7f54950 93 64 F9 29 18 74 3F 39-01 00 00 00 00 00 00 00 00 othi) t79	*
	c7f54960 Å3 64 F9 29 28 01 66 34-01 00 00 00 D8 F0 F8 29 £dù) (·f4····⊘8ø)	
	c7f54970 B3 64 F9 29 10 5F B6 34-B0 F0 F8 29 00 00 00 00 00 dù) . 44 880	
	c7f54980 C3 64 F9 29 00 6E 3F 39−30 00 00 05 2 00 65 00 <mark>Ådù) n?90···R·e·</mark>	
	c7f54990 63 00 65 00 69 00 76 00-65 00 64 00 20 00 77 00 <mark>creirvedt we</mark>	
	c7f549a0 69 00 74 00 68 00 20 00-31 00 48 00 4B 00 51 00 <mark>i.t.h1.H.K.Q.</mark>	
	c7f549b0 4C 00 69 00 6F 00 6D 00-7A 00 4C 00 57 00 73 00 L·i·o·m·z·L·W·s·	
	c7f549c0 77 00 42 00 7A 00 61 00-41 00 34 00 35 00 32 00 <mark>w·B·z·a·A·4·5·2·</mark>	
	c7f549d0 63 00 62 00 56 00 69 00−53 00 75 00 31 00 6E 00 <mark>c·b·V·i·S·u·1·n·</mark>	
	c7f549e0 66 00 57 00 70 00 31 00−48 00 76 00 00 00 00 00 00 <mark>f.W.p.1.H.v</mark>	
	c7f549f0 33 65 F9 29 00 6E 3F 39-2A 00 00 00 53 00 65 00 3eù) n?9*5.e.	
	c7f54a00 6E 00 74 00 20 00 74 00-6F 00 20 00 31 00 33 00 n·t··t·o··1·3·	
	c7f54a10 6D 00 69 00 41 00 70 00-34 00 33 00 5A 00 65 00 m·1·A·p·4·3·Z·e·	
	c7f54a20 57 00 4A 00 35 00 64 00-75 00 67 00 38 00 58 00 W·J·5·d·u·g·8·X·	
	c7f54a30 74 00 68 00 32 00 38 00-55 00 64 00 34 00 52 00 <mark>t·h·2·8·U·d·4·R·</mark>	
	c7f54a40 4A 00 57 00 71 00 36 00-36 00 4E 00 75 00 31 00 J W q 6 6 N u 1	
	c7f54a50 93 65 F9 29 18 74 3F 39−01 00 00 00 50 69 0F 00 •eù) ·t?9····Pi··	

Figure 13. Received with and sent to addresses can be found in RAM. This can provide examiners with recent additional addresses that are associated with the account.

Capturing Bitcoin network traffic contained a great amount of transaction information. The latest version of Wireshark (Version 1.10.0) contains a Bitcoin protocol – greatly assisting the examiner in analyzing Bitcoin network traffic. The Bitcoin protocol, operating from port 8333, uses many message types, the most important for examinations being inv and tx¹⁶. The inv messages are typically the first sign of a transaction and an indication that the transaction has initially been accepted by the network. inv represents a node on the network advertising the knowledge of either a transaction (type 1) or a block (type 2) (Fig. 14). These nodes will use the same Payload checksum for this specific object. For message type 1 inv, the transaction hash is represented in Big Endian stored in the Data hash field.

📕 rec	ievetradefro	mblockwallet.p	cap [Wiresha	rk 1.10.0 (SVN	I Rev 49790 fro	m /trunk-1.10	0)]						□ <u> </u>
<u>F</u> ile	<u>E</u> dit <u>V</u> iew	<u>G</u> o <u>C</u> apture	e <u>A</u> nalyze <u>S</u>	Statistics Tele	ephon <u>y T</u> ools	<u>I</u> nternals	<u>H</u> elp						
0 (۰ 🖌 🔳	1 E	🖿 🗶 🖉	् 🗢 🔿	<table-cell></table-cell>			🖭 🌌 🗵	R 🕺 🚺	B			
Filter:	Filter: bitcoin.inv Expression Clear Apply Save												
		:40.974414 :41.056922			Destination 192.168.1		Protocol Bitcoin Bitcoin	Length Inf 115 in 115 in	IV				^
98 99	354 15:52 354 15:52	:41.156725 :44.395766	107.170. 83.40.36	211.138 .156	192.168.1 192.168.1	L39.141 L39.141	Bitcoin Bitcoin	115 [1 200 ac	CP Retrans Idr, inv		, inv		=
99	73 15:52	:44.494285 :45.447889 :45.546750	98.186.1	67.15	192.168.1 192.168.1 192.168.1	L39.141	Bitcoin Bitcoin Bitcoin	115 ir	CP Retrans V CP Retrans		, addr, in , inv	nv	
100	71 15:52	:48.546980 :48.646766 :03.461043	85.214.1	00.177	192.168.1 192.168.1 192.168.1	L39.141	Bitcoin Bitcoin Bitcoin	115 ir	IV CP Retrans		, inv		
		:03.558849			192.168.1		Bitcoin		CP Retran:	smission]	, inv		
	Command Payload Payload Conventory Count: Inventory Count: Inventory	agic: 0xf9 name: inv Length: 37 checksum: y message 1 pry vector : MSG_TX (0x5477dd01		b34902c84f	7e07dbba9	9Fd36a5b◀	🗲 Big En	dian trans	action ha	sh		A E
0040 0050 0060 0070	00 65 6 8b 8d 2 fa f0 1 00 00 0 02 c8 4 1d b0 5	Lc bf 00 0 00 00 00 0 00 dd bb b 4f 7e 07 d 59	0 80 06 4 c 4f 7a c 0 f9 be b 0 25 00 0 6 b1 00 2 b ba 9f c	te bf 6b a d7 6e 2c 7 o4 d9 69 6 00 00 54 7 25 be 07 b d3 6a 5b 5	0 08 00 45 a d3 8a c0 e f6 5a 50 e 76 00 00 7 dd 0f 01 4 3c cc b3 2 3a 44 7c	a8 .e [°] 18 00 01 49 f90 Y		ZP. D .					
• 💅	File: "G:\Su	mmer Project\	Windows 7\Wi	indo Packe	ts: 11038 · Displa	ayed: 14 (0.1%	6) · Lo Profile	: Default					

Figure 14. The inv message represents a node (107.170.211.138) acknowledging an object – typically a transaction (type 1) or a block (type 2). All inv and getdata messages for this particular transaction will share the same payload checksum.

The inv message is followed by a getdata message, which contains a matching payload checksum to

its parent inv and the transaction hash. The getdata message is a response from the host to the node

¹⁶ Wiki, Bitcoin. Protocol specification. < https://en.bitcoin.it/wiki/Protocol_specification>.

acknowledging the transaction (Fig. 15).

📕 recie	vetradefromblo	ckwallet.pcap [Wireshar	k 1.10.0 (SVN Rev 4	9790 from /trunk-1	10)]	1.00	A CONTRACTOR OF		
<u>F</u> ile <u>E</u>	dit <u>V</u> iew <u>G</u> o	<u>Capture</u> <u>A</u> nalyze <u>S</u> t	tatistics Telephony	<u>Iools</u> Internal	s <u>H</u> elp				
0 0) 🛋 🔳 🔬	🖹 🖹 🗶 🛃	् 🔶 🔿 🎝	7 ⊻ ■		🖭 🔐 🗹 🕵 💥	B		
Filter:	bitcoin.getdata			- Exp	ression Clear Ap	ply Save			
No.	Time	Source		ination	Protocol	Length Info			A
		.975160 192.168.1 .241796 192.168.1		0.162.35.196 0.162.35.196	Bitcoin Bitcoin	115 getdata 115 getdata			
		.792008 192.168.1		9.162.35.196	Bitcoin	115 getdata			
		.679627 192.168.1		3.77.19.18	Bitcoin	115 getdata			-
٠ -									4
🗄 Fra	me 9847: 1:	15 bytes on wire	(920 bits), 1	15 bytes capt	ured (920 bit:	;)			
						cf:90 (00:50:56:f			
							6 (109.162.35.196)		
			Src Port: 49	165 (49165),	Dst Port: 833	3 (8333), Seq: 521	.7, Ack: 5553, Len: 61	L	
	coin proto	col c: 0xf9beb4d9							
	ommand nam								
	ayload Len								
P	ayload che	cksum: 0x5477dd0f							
🗆 G	etdata mes	sage							
	Count: 1								
	Inventory								
	Type: M	5 <mark>G_TX (1)</mark> sh: ddbbb6b10025b	e07b43cccb349	02c84f7e07dbb	a0fd36a5h	 Big Endian tra 	insaction hash		
	Daca na.	311. 0000000100230	e07 04 3 C C C D 34 30	0200417807000	a91030a50	Dig Endian da			
0030	fa f0 dd 1	F3 00 00 f9 be b	4 d9 67 65 74	64 61 74 .		lat			
0040	61 00 00 0	00 00 00 25 00 0 dd bb b6 b1 00 2 7e 07 db ba 9f d	0 00 54 77 dd	0f 01 01 a	%Ťw				
0050	02 c8 4f	7e 07 db ba 9f d	3 6a 5b 52 3a	44 7c f9	%< .0~j[R:0				E
0070	1d b0 59				.Y				*
e 🕅	Count (hitcoin (getdata.count), 1 byte	Packets: 110	88 . Displayed: 4 (0.0	1%) Profile: Def	ault			
	count (bitcoini	geraatareount), 1 byte	. sekets, 110.	o bispisyed. 4 (0.0	noy an erome ben				

Figure 15. An inv will result in a getdata reply from the host (192.168.139.141) to the node (109.162.35.196).

The node then sends a tx message with the transaction details corresponding to the transaction

hash in the previous inv and getdata messages. The tx message will have a payload checksum matching

that of the first 4-bytes of the inv and getdata Data hash value. The tx message contains most of the

information publically available on the blockchain (Fig. 16).

🖌 recievetradefromblockwallet.pcap [Wireshark 1.10.0 (SVN Rev 49790 from /trunk-1.10)]	_ 🗆 X
File Edit View <u>G</u> o <u>C</u> apture Analyze Statistics Telephony Iools Internals <u>H</u> elp	
9 🖲 🧉 🔳 📶 ڬ 😂 이 🗢 🗢 🖓 🛧 🗐 🗐 🗐 이 오 오 이, 💟 🎬 🖄 🥵 💥 🗒	
Filter: bitcoin.tx Expression Clear Apply Save	
No. Time Source Destination Protocol Length Info	*
9858 15:52:41.236868 109.162.35.196 192.168.139.141 Bitcoin 302 tx	
9888 15:52:41.790720 109.162.35.196 192.168.139.141 Bitcoin 304 tx	*
	,
🗆 Bitcoin protocol	*
Packet magic: 0xf9beb4d9	
Command name: tx	
Payload Length: 224	
Payload checksum: 0xddbbb6b1	
TX message Transaction version: 1	
Input Count: 1	
□ Transaction input	
Previous output	
Hash: efe69776807a521d3b6f4bd5a11d16128aa14d763e1fbcbb 🛶 🛛 Big Endian previous transaction hash	
Index: 0	=
Script Length: 139	-
signature script: 4830450220101c7401a8616d41cc62b451632da959b7e3c6	
Sequence: 4294967295	
output Count: 1 - Number of Outputs	
Transaction output value: 440000 Amount Sent	
Script Length: 25	
script: 76a914b2fcfbad6ce94a81f4c288157fd48a6f7c78773a88 🗲 160-bit hash of Public Key	
Block lock time or block ID: 0	*
0030 fa f0 92 4a 00 00 f9 be b4 d9 74 78 00 00 00 00	*
0050 00 00 01 ef e6 97 76 80 7a 52 1d 3b 6f 4b d5 alv. zR.;ок	
0060 1d 16 12 8a a1 4d 76 3e 1f bc bb 1a 94 fd ed 43Mv>C	
0070 1f e1 02 00 00 00 08b 48 30 45 02 20 10 1c 74	-

Figure 16. The tx message from the node (109.162.35.196) is in reply to the getdata message from the host (192.168.139.141). This message will have a payload checksum matching the first 4-bytes of the data hash in the previous inv and getdata messages. The tx message contains most of the information that would be publically viewable on the blockchain.

The Input and Output Count fields indicate the number of addresses involved. The "Previous output Hash" is the previous transaction hash in big Endian of each input address. Each output address has the corresponding amount sent and the address sent to in the "Value" and "Script" fields, respectively. It should be noted that the actual value sent and the value observed in Wireshark is a difference of 1 x 10⁸. Additionally, the "Script" value must first be processed to arrive at the address as seen on the blockchain. This process involves taking the 25-byte "Script" value from Wireshark and removing the first 3-bytes and the last 2-bytes and running the remaining 20-bytes through a series of hashes (Fig. 17). The 5-bytes removed are simply Bitcoin protocol artifacts that are not necessary to decode the address.

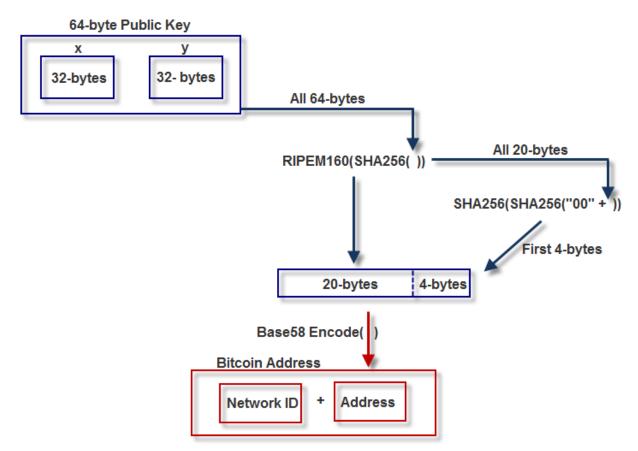


Figure 17. A more detailed version of Fig. 6 starting with the generated ESDCA public key, which is made up of its (x, y) position on the curve. The Script value from Wireshark is already the 20-byte product from the RIPEM160 and SHA256 hash of the public key. After removing the first 3-bytes and last 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.

There are many artifacts generated when a user uses MultiBit 0.5.18. The multibit.log and

network captures contain the most valuable information to an examiner. Both of these will contain

information needed to identify transactions on the user's computer. The multibit.log,

RecentFileCache.bcf, and IzPack files can also help with creating a user activity timeline from daily

activity to date of uninstallation. The MultiBit lnk files can pinpoint on which user account the

application is installed. These artifacts not only assist the examiner with the MultiBit client, but in the

case of the network captures, any client or even cryptocurrency using the Bitcoin protocol.

Litecoin – Litecoin 0.8.7.2

The Litecoin wallet software shares many similar artifacts with MultiBit. The main folder for Litecoin

is in the C:\Users\[User]\AppData\Roaming\Litecoin and /\home\[User]\.litecoin directories for

Windows 7 and Ubuntu 14.04, respectively. The wallet.dat file located in this directory contains

references to addresses received on and sent to (Fig. 18).

Home 🕆 Entries 🛈 Bookmarks 🔍 Search Hits 🗟 Records 🖾		Name	Description	File Type	File Ext	File Created	Entry Modified				
C Home Hash Properties	1	🗋 wallet.dat	File, Archive, Not Indexed	Data ASCII & Binary	dat	06/24/14 10:14:08AM	07/01/14 12:32:06PM				
- 🕞 ? LeUlulrCRhxtWE88zn62ZXMHodUM9vXkUuA 🔦 🖸 2 🗋 debug.log File, Archive, Not Indexed Log log 06/24/14 10:14:08AM 07/01/14 12:32:07PM											
-D_ ? L2uRom68Kat4wUcjpWxrXuGoGmJ51D922 + + + III											
🔚 Text 🐙 Hex 🎧 Doc 🦙 Transcript 🔤 Picture 🔄 Report 🖸 Console 🕲 Details 🖉 Output 📿 Lock 📿 Codepage 📿 0/81951											
49170 - ·㷷·Ô···t·Q···à··(-I··ô·À· ··`·4···Ô···t·Q·· 49319 - T··ô·À· ·`·4···Ô··· ·L···ã···· ·x·L·, ·æ·Ă·· 49468 - T····T·····t······	·p ·< ·	, ø·è·'·¤·p·`·, ···	ÈØ¤ `₽··Ø·È·	· · P · @ · · · ü · È · , · · t	-0-9-	üì, "td0 ì	Ü" dT -				
49468 Ü·İ· · ·T·D····Ì·Ma· ·x·D·4···3·Ma·m·x·h·4·\$·3·à·m· ·h·X·\$··à·Đ· · ·X·H····Đ·À· · ·H·8···ô·À·ø·Ä·····Second trade to Hivewalletrs(···name"Lec3FtnCynVfbXGE 49617 49617!··òCŢĨd£;ñ5·à·f5À·····Money from Ubuntu Wallet(···name"LEUDUtiWK6bJHSWJZmqyYTFS9cG4kDTmdn····Second trade to Hivewalletrs(···name"Lec3FtnCynVfbXGE 49766F2tqMW74rcUMA9WHB											
4991515>;δ @S											

Figure 18. The wallet.dat file contains references to addresses received on and sent to. However, other transaction details such as a timestamp or an amount are not present.

On Windows, the Litecoin and Uninstall Litecoin Ink files contained the SID of the user account they

😂 Home 🔚 Entries 💭 Bookmarks 🔍 Search Hits 🗠 Records 🖾 🕩		Name	Description	File Type	File Ext	File Created	Entry Modified				
🗄 Home 🕬 File Extents 🔒 Permissions 👸 References 📝 Hash ()	103	📾 Command Prompt.Ink	File, Archive	Windows File Shortcut	Ink	12/12/12 12:00:41PM	06/17/14 05:33:25PM				
B-D tries	104	🖙 Notepad.lnk	File, Archive	Windows File Shortcut	Ink	12/12/12 12:00:41PM	06/17/14 05:33:25PM				
🖕 D 🔽 🙍 untitled	105	📾 Litecoin.lnk	File, Archive	Windows File Shortcut	Ink	06/24/14 10:13:07AM	07/01/14 11:52:18AM				
🖨 🖶 🗔 🥨 C	106	∞ Uninstall Litecoin.lnk	File, Archive	Windows File Shortcut	Ink	06/24/14 10:13:07AM	07/01/14 11:52:21AM				
🕀 🍯 💭 🧰 \$Extend	107	©@ Help.Ink	File, Archive	Windows File Shortcut	Ink	12/12/12 12:00:41PM	06/17/14 05:33:26PM				
🖬 🍯 🍘 \$Recycle.Bin	108	Sea Internet Explorer (6	File, Archive	Windows File Shortcut	Ink	12/12/12 12:01:19PM	06/17/14 05:33:25PM				
🕀 🍯 💭 🧰 Boot	1	uncence explorer (orm	They Arenive	Thindows the briotecar	ii iis	12,12,12,12,12,01,13,14	00/17/1100:00:2010				
Decuments and Cettings											
📔 Text 🧔 Hex 🎡 Doc 🐏 Transcript 🖾 Picture 📃 Report 💽 Conse	ole 🔞 Dei	tails 🗊 Output 🛛 Lock	🗌 Codepage 🛛 0/81951								
0000LÀF /£.ÀË.öj¶ Ï /£.Ài											
0150 ····ï¾î: ·ØD£q*···÷·····R····P·r·o·g·r·a	·m· ·F·i	i.l.e.s(.x.8.6.).	··@·s·h·e·l·l·3·2·.	·d·1·1·, ·-·2·1·8·	1.7.	$\cdots V \cdot 1 \cdots \cdot @\texttt{D} \texttt{M} q \cdots$	Litecoin ··>····				
0300 · ï%2D£q2D¤q*···sê···········L·i·t·e·c·o·i											
0450 ·e·x·e·····											
$0600\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot\cdot$											
0750 ·i·t·e·c·o·i·n····· *···Á······ ï@Z û üK JÀd	òà¹ú∣Á··	·· ··· ·· ···1SPSâ	XF%L8C»ü· & mÎm····	····· <mark>S·-·1</mark>	L 5 -	2 -13 -4 -0 -8 -8	·1 ·1 ·4 ·5 ·2 ·- ·3 ·				
0900 3 - 1 - 0 - 7 - 6 - 3 - 1 - 7 - 8 - = - 3 - 6 - 3 - 0 - 4 - 0 - 0 - 8 - 1 - = - 1 - 0 - 0 - 0	• • • • • • • •	· · · · · · · · X · · · · · win	-gb2pkaq230h· ·âî;	A å3 0¢7 ∙o •©ûâ	i∙«1 · ·) þÆ ·âî; A å3 0	¢7 ·o ·©ûã·«1··				
1050) p£	• • • • • • • •			•••••		• • • • • • • • • • • • • • • • • • • •	•••••				
1200				•••••		• • • • • • • • • • • • • • • • • • • •					
1350	• • • • • • • •	• • • • • • • • • • • • • • • • • • • •	••••••	•••••		•••••	•••••				
1500 ScanQ end Va						0, 0x0000000)}					
1650Committing file queue flq: {_commit_s			CommitQ DelNo								
		nit_file_queue exit		Registering CoIn							
		lass installer for '			Insta	ller 1: Enter 11:					
2100 dvi: CoInstaller 1: Exit dvi:		Default installer: E		inf:		Opened PNF: 'c					
2250m32\driverstore\filerepository\machine.inf_amd64_ne							all Inf Section				
		Install Inf Section									
			ler 1: Enter 11:58:	25.158 dvi:							
2550 {DIF_DESTROYPRIVATEDATA} 11:58:25.158 dvi:							oInstaller 1: E				
2550 {DIF DESTROYPRIVATEDATA} 11:58:25.158 dvi: 2700xit dvi: Default installe	er: Ente	er 11:58:25.158	dvi:	Default i		ler: Exit d	vi:				
2550 {DIF_DESTROYPRIVATEDATA} 11:58:25.158 dvi:	er: Ente	er 11:58:25.158	dvi: Co-instal	Default i		ler: Exit d					

were installed on (Fig. 19).

Figure 19. The Litecoin Ink files contain the SID belong to the user account which the application was installed on.

Litecoin has a debug.log file, in the directory previously mentioned, that functions in a similar manner to the multibit.log file. It contains information relating to user usage of the application and transaction data (Fig. 20). As it is a log file and can be overwritten with heavy usage, only recent usage and transaction data will be contained. All timestamps appear to be in UTC. Transactions sent from the

wallet begin with "CommitTransaction". There are received transaction hashes present in debug.log,

however, they do not contain full transaction information like multibit.log.

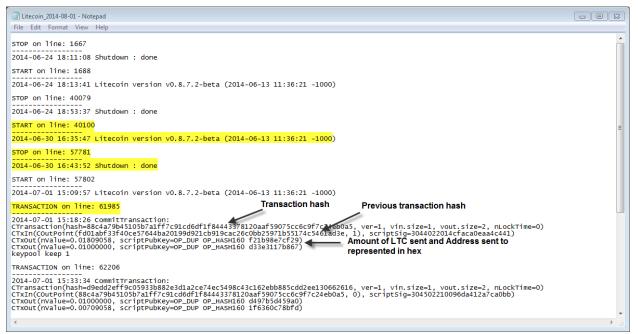


Figure 20. Parsing the debug.log file can be useful to an examiner attempting to determine recent user activity and transactions.

Litecoin registry keys only had values relating to system information. Litecoin keys were found in

SYSTEM and SOFTWARE hives. The following keys were associated with Litecoin installation:

- HKLM\SYSTEM\Classes\litecoin
- HKLM\SYSTEM\Wow6432Node\Litecoin
- HKU\[User SID]\SOFTWARE\Microsoft\Windows\Uninstall
- HKU\[USER SID]\SOFTWARE\Litecoin

When Litecoin was uninstalled, all the previously mentioned artifacts were still present with the

exception of the registry keys and lnk files. Unlike MultiBit, uninstalling Litecoin did not lead to the

creation of an object that could date that activity.

String analysis of the VM VMEM files led to similar results seen in MultiBit as there were

multiple references to addresses that were sent to during that user session (Fig. 21).

Q AccessData FTK Imager 3.1.3.2	AND ADDRESS OF THE OWNER ADDRE	
<u>F</u> ile <u>V</u> iew <u>M</u> ode <u>H</u> elp		
🎄 🎎 🗣 🕰 🚖 🖂 🖬 🖬 🖬 🚑 🖂 🗆 🚥	📴 🥄 🗋 🖹 🖻 🚾 😹 🐮 🎗 🖕	
Evidence Tree ×	ad3a0470 74 0A 64 0A 30 0A 20 0A-EC 09 DC 09 A8 09 98 09 t.d.o. i.Ü."	
	ad3a0480 64 09 54 09 20 09 10 09-DC 08 CC 08 98 08 88 08 d·T·····Ü·Ì·····	
	ad3a0490 54 08 44 08 10 08 00 08-CC 07 BC 07 88 07 78 07 T·D·····Ì·¾···x·	
	ad3a04a0 44 07 34 07 00 07 F0 06-BC 06 AC 06 78 06 68 06 D.4···ð·h·-·x·h·	
	ad3a04b0 34 06 24 06 F0 05 E0 05-AC 05 9C 05 68 05 58 05 4 \$ \$ ah.X.	
	ad3a04c0 24 05 14 05 E0 04 D0 04-9C 04 8C 04 58 04 48 04 \$X.H.	
	ad3a04d0 14 04 04 04 00 03 C0 03-8C 03 7C 03 48 03 38 03 ····Đ·À·····H·8· ad3a04e0 04 03 F4 02 C0 02 F8 01-C4 01 00 00 00 00 00 00 ····Đ·À·@·Ä······	
	adjacijo je do ol je	
	ad3a0520 03 7E D2 C7 CF D8 A3 3B-F1 35 04 EA B2 35 C3 2A OCIØ#; n5 - e* 5A*	
	ad3a0530 19 00 01 18 4D 6F 6E 65-79 20 66 72 6F 6D 20 55 ····Money from U	
	ad3a0540 62 75 6E 74 75 20 57 61-6C 6C 65 74 28 00 01 04 buntu Wallet(
	ad3a0550 6E 61 6D 65 22 4C 54 62-44 75 74 69 57 4B 36 62 name"LTbDut1WK6b	
	ad3a0560 4A 72 48 73 57 4A 5A 6D-71 79 59 54 46 53 39 63 JTHSWJZmqyYTFS9c	
	ad3a0570 47 34 6B 44 54 6D 64 6E-1B 00 01 1A 53 65 63 6F G4kDTmdn. Seco	
	ad3a0580 6E 64 20 74 72 61 64 65-20 74 6F 20 48 69 76 65 nd trade to Hive	
	ad3a0590 77 61 6C 6C 65 74 72 73-28 00 01 04 6E 61 6D 65 walletrs(name	
	ad3a05a0 22 4C 65 63 33 46 74 74-6E 43 79 6E 56 66 62 58 "Lec3FttnCynVfbX	
Hex Value Interpreter ×	ad3a05b0 47 45 50 61 74 71 6E 57-54 34 72 61 55 31 4D 41 GEPatqnWT4raU1MA ad3a05c0 39 4D 48 42 18 00 01 17-46 69 72 73 74 20 54 72 9MHB ····First Tr	
· · · · · · · · · · · · · · · · · · ·	ad3a050 61 6E 73 4D 40 47 42 10 00 01 1/40 69 72 73 74 20 54 72 95 76 77 11 15 17	
Type Size Value	adjaosto lo 2/3 00 10 4 6E 6 6 65-22 4C 65 575 4C 72 43 (···name"LeOuldC	
signed integer 1-8	adaabsto 52 68 78 74 57 74 58 42-74 6E 36 32 5A 54 4D 48 RhatteeBaza622XMH	
unsigned integer 1-8 FILETIME (UTC) 8	ad3a0600 6F 63 55 4D 39 76 58 6B-55 75 41 00 2E 00 01 3E ocUM9vXKUuA>	
FILETIME (OCC) 8 FILETIME (local) 8	ad3a0610 3B 01 00 F5 87 A9 53 00-00 00 02 1 02 B2 A5 E0 ; ··Õ·⊕S····!·*¥à	
DOS date 2	ad3a0620 06 65 9F E9 AC B5 11 05-7D 55 D0 EF C1 A8 1D 84 .e. (e.,µ)UĐIĂ	
DOS time 2	ad3a0630 62 BB 34 2E 2C FD 06 A9-C8 8E 2C B9 1D 00 00 00 b≫4.,ý ⊕È.,*····	
time_t (UTC) 4	ad3a0640 0D 00 01 04 70 6F 6F 6C-35 00 00 00 00 00 00 00 00pool5	
time_t (local) 4	ad3a0650 2E 00 01 3E 3B 01 00 F5-87 A9 53 00 00 00 02 1>;	
	ad3a0660 03 88 58 92 AE D2 81 58-61 A6 DA 32 32 33 EB D7 · X @O Xa (Ú238×	
	ad3a0670 04 10 8C 4F F2 AC DE 86-FC 03 61 8C 97 A1 8A 2A ···O∂→F·ù·a··;·*	
	ad3a0680 69 00 00 00 00 00 01 04-70 6F 6F 6C 34 00 00 01 01 00 01 01 00 01 00 00 00 00	
	adsadoso 00 00 00 02 00 01 35-35 01 00 F5 67 A 9 53 00 (·················s). adsadosa 00 00 02 1 03 B9 1 A 7C-EB E0 AE 25 55 05 5F C3 (···································	
	adsardobo de 92 97 3A 37 54 DE 9F-15 B7 FC 9 1B 6A E5 78 N·:7TF····É:-jÅx	
۰ III ۲	ad3a06c0 2A C1 0D 6B 7E 00 00 00-0D 00 01 04 70 6F 6F 6C *A.k~pool	
Byte order: 🕟 Little endian 🔿 Big endian	ad3a06d0 33 00 00 00 00 00 00 00 -2E 00 01 3E 3B 01 00 F5 3	
	ad3a06e0 87 A9 53 00 00 00 00 21-03 AF 83 AD D8 C4 A1 53 -⊕S····!· ⁺ ·-ØÄ;S	-
Properties Hex Value Int Custom Conte	Cursor pos = 2906260960; log sec = 5676290	
For User Guide, press F1		NUM //

Figure 21. String searching in memory is another source to observe recent transactions between the last shutdown of the machine.

Litecoin has similar network traffic as Bitcoin. Unfortunately, there is currently no built in protocol as of Wireshark (Version 1.10.0). Wireshark categorizes Litecoin traffic under the TCP protocol. Litecoin traffic operates on port 9333 and contains many of the same messages as Bitcoin including: inv, getdata, and tx. As there is no built in protocol an examiner must manually parse the litecoin traffic (Fig. 22). The inv and getdata messages are relatively straightforward to parse, normal packets will be 115-bytes in length and contain the hash of either a transaction or block in big endian.

	inv		getdata		tx
Byte offset	Item	Byte offset	Item	Byte offset	Item
0-53	Packet Structure	0-53	Packet Structure	0-53	Packet Structure
54-	Data	54-	Data	54-	Data
54-57	Packet Magic	54-57	Packet Magic	54-57	Packet Magic
58-69	Command Name (69:6e:76)	58-69	Command Name	58-69	Command Name (74:78)
70-73	Payload Length	70 70	(67:65:74:64:61:74:61)	70-73	Payload Length
74-77	Payload checksum	70-73	Payload Length	74-77	Payload checksum
78	Count	74-77	Payload checksum	-	-
79-82	Type (01 - TX, 02 - Block, 03 -	78	Count	78-81	Transaction Version
10-02	Uknown)	79-82	Type (01 - TX, 02 - Block, 03 -	82	Input Count
83-114	Data hash		Uknown)	83-262*	Transaction Input *Variable length - 180 bytes in this example
		83-114	Data hash	263	Output Count
				264-297	Transaction Output *Amount of tx output fields dependent on output count
				297-301	Block lock time or Block ID

Figure 22. 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.

The tx message is more difficult to parse due to variable fields (Fig. 23). The Transaction Input

field contains 36-byte Previous output (32-byte transaction hash and 4-byte index), 1-byte Script

Length, variable length Signature Script, and 4-byte Sequence fields. Typically, there is only one

Transaction Input field, however it is more common to see multiple Transaction Output fields. The

number of Transaction Output fields will depend on the value of Output Count. Transaction Output

contains an 8-byte Value (represents the 1 x 10⁸ amount sent), 1-byte Script Length, 25-byte Script

fields.

recievetradefromUbuntu.pcap [Wireshark 1.10.0 (SVN Rev 49790 from /trunk-1.10)]			_ _ X
<u>File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help</u>			
◉ ◉ ∡ ■ ∡ ⊨ ≞ ೫ ≋ ! < 수 ㅎ ৯ ₮ ⊻ । 🗐 🗐 ! < < <	🎬 🗹 🍢 💥 🛛 🧱		
Filter: tcp.port == 9333	Save		
	ngth Info		*
68 11:58:33.941885 109.228.164.67 192.168.139.144 TCP 69 11:58:33.957882 192.168.139.144 188.138.197.50 TCP		Seq=117 Ack=62 Win=64240 Len= Seq=1 Ack=1 Win=64952 Len=55	250
09 11:38:33.95/882 192.108.139.144 188.138.197.50 TCP	109 42993 > 9333 [PSH, ACK]	Seq=1 ACK=1 WIN=64952 Len=55	
			,
B Frame 68: 504 bytes on whe (2452 bits), 304 bytes captured (2452 bits) B Ethernet II, Src: Vmware_f6:cf:90 (00:50:56:f6:cf:90), Dst: Vmware_df:b6:(03 (00:0c:29:df:b6:03)		
H Internet Protocol Version 4, Src: 109.228.164.67 (109.228.164.67), Dst: 19	92.168.139.144 (192.168.139.		
Transmission Control Protocol, Src Port: 9333 (9333), Dst Port: 36235 (362)	235), Seq: 117, Ack: 62, Len	: 250	
Data (250 bytes) Data: fbc0b6db747800000000000000000000000000000000000			
[Length: 250]			
0000 00 0c 29 df b6 03 00 50 56 f6 cf 90 08 00 45 00)P VE.	Alternating color scheme to	differentiate items listed	
0010 01 22 c0 d0 00 00 80 06 1a a5 6d e4 a4 43 c0 a8 ."			
0030 fa f0 11 7b 00 00 fb c0 b6 db 74 78 00 00 00 00 {	1. Packet Magic	12. Output Count	
0040 00 00 00 00 00 00 e2 00 00 00 5c a6 a0 ec 01 00	2. Command Name	-	
0060 d6 0f 27 24 ca 98 14 f8 8d ec a8 5d 80 a6 2e ed\$]	3. Payload Length	Transaction Output	
0070 e4 d9 c5 00 00 00 00 6b 48 30 45 02 20 66 f6 0bk HOE. f 0080 a7 c9 6c 47 a7 15 fb a6 7a ec 26 57 73 7d 5d 96]G z.&ws}].	 Payload Checksum 	13. Value	
0090 09 54 79 b5 0c 0d 2e 9c d5 0e 47 bf 94 02 21 00 .TyG!	5. Transaction Version	14. Script Length	
00a0 d5 54 e6 d0 8b 86 ca b5 71 b8 58 e7 f6 1b e7 ba	6. Input Count	15. Script	
00c0 01 21 03 52 0a 77 ee 8e f6 90 80 8e c0 35 a7 69 .!.R.w5.i			
00d0 d5 d8 07 2c 0d 5c 79 0e e3 62 d3 c8 3e c0 b8 94,.\y. b> 00e0 7f 54 a0 ff ff ff ff 02 79 5a 1a 0d 00 00 00 00	Transaction Input	Transaction Output	
00f0 19 76 a9 14 ad ab e6 1b 2e 0a 43 46 86 36 e6 49 .v	7. Previous Output - Hash	16. Value	
0110 00 00 19 76 a9 14 69 0c 1e a6 90 02 fb 12 51 68v.iQh	8. Previous Output - Index	17. Script Length	
0120 4d 77 76 7e a8 da 19 b4 b5 1e 88 ac 00 00 00 00 Mwv~	9. Script Length	18. Script	
	10. Script Signature 11. Seguence	19. Block Lock Time or Block	חו
🖨 💓 Eller "CA Commerce Devices) Wite davor 7) Wite da	n. sequence	13. BIOCK LOCK TIME OF BIOCK	
🔴 💆 File: "G:\Summer Project\Windows 7\Windo Packets: 1382 · Displayed: 117 Profile: Default			

Figure 23. 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 "L").

Installation of the Litecoin wallet leaves behind many pertinent artifacts on the user's computer.

An examiner can determine which account the program is installed on, recent transactions, and

general usage of the application. Even after uninstalling the application, these artifacts are still

observable. Unlike MultiBit, there was no unique file created when uninstalling the application.

Darkcoin - Darkcoin 9.11.6

Darkcoin shares many similarities to Litecoin. Perhaps because the two wallets examined both utilize

the Qt application framework. Installation did not create lnk files to analyze. The wallet.dat and

debug.log files have similar artifacts and can be found in C:\Users\[User]\AppData\Roaming\Darkcoin

and /\home\[User]\.darkcoin directories on Windows and Ubuntu, respectively. The wallet.dat file

contains references to addresses belong to the wallet and those that were traded to (Fig. 24).

Cases Mash Sets X	Ta	ible 📃 Report 💷 Gallery	🕑 Timeline 📮 Disk 🔧 Cod	e						
Home Entries DBookmarks 🔍 Search Hits 🖏 Records 💭 Devices 💱 🕩		Name	Description	File Type	File Ext	File Created	Entry Modified			
🚰 Home 🕫 File Extents 🔒 Permissions 🕅 References 📝 Hash Properties	2 6	debug.log	File, Archive, Not Indexed	Log	log	06/30/14 08:48:28AM	07/17/14 04:07:57PM			
Roaming A	7	🗅 peers.dat	File, Deleted, Overwritten, A	Data	dat	06/30/14 09:13:09AM	06/30/14 09:13:09AM			
-DI Con DarkCoin	28	peers.dat	File, Archive, Not Indexed	Data	dat	07/17/14 04:07:57PM	07/17/14 04:07:57PM			
B-D C blocks	2 9	🗋 wallet.dat	File, Archive, Not Indexed	Data	dat	06/30/14 08:48:28AM	07/17/14 04:07:57PM			
	10	💫 wallet.dat.del.0858	File, Deleted, Archive, Not In		085	06/30/14 08:48:28AM	06/30/14 09:35:31AM			
database 🗸	•									
🔚 Text 🐙 Hex 🍙 Doc 🔚 Transcript 🔛 Picture 📃 Report 💽 Console 🗃 Deta	ils 🗊 o	utput 🗌 Lock 🗌 Codepi	age 🗌 0/81262							
48272å 14j\$p.96]**#82‡8*732 Yr.5.1+126Åx 1.9.C***********************************	10 Lo Wallet.dat.del.0656 File, Deleted, Archive, Not In 06/30/14/08:46:26AM 06/30/14/09:35:31AM									
C\Users\ AppData\Roaming\DarkCoin\wallet.dat			•							

Figure 24. The wallet.dat file contains references to both addresses belonging to the wallet and addresses that were sent to in transactions.

The debug.log file has the same format and type of information as the Litecoin variant.

CommitTransaction indicates transactions sent from the Wallet. Application start and stop times are

noted by "Darkcoin version" and "Shutdown : done" lines. (Fig. 25).

Cases 🧏 Hash Sets 🗙 🗙	i 🔡 Ta	able 📃 Report 🗵 Gallery	🛛 🎯 Timeline 일 Disk 🏄 Cod	le			
🕒 Home 🏗 Entries 💭 Bookmarks 🔍 Search Hits 🥰 Records 💭 Devices 💱 🕩		Name	Description	File Type	File Ext	File Created	Entry Modified
🔚 Home 📴 File Extents 🔒 Permissions 🛱 References 🏼 Hash Properties	23	🚞 chainstate	Folder, Not Indexed			06/30/14 08:48:28AM	07/17/14 03:53:03PM
	4	词 database	Folder, Deleted, Not Indexed			06/30/14 08:58:06AM	06/30/14 09:28:21AM
	2 5	🗋 db.log	File, Archive, Not Indexed	Log	log	06/30/14 08:48:28AM	06/30/14 08:48:28AM
D D D D D D D D D D D D D D D D D D D	2 6	🗋 debug.log	File, Archive, Not Indexed	Log	log	06/30/14 08:48:28AM	07/17/14 04:07:57PM
-D C hainstate	7	🞝 peers.dat	File, Deleted, Overwritten, A	Data	dat	06/30/14 09:13:09AM	06/30/14 09:13:09AM
-D - database	•	III					
🔚 Text 🐙 Hex 🎧 Doc 🧐 Transcript 🔝 Picture 📃 Report 💽 Console 🔞 Deta	ils 🗊 o	Dutput 🗌 Lock 🗌 Codep	age 🗌 0/81262				
000147378f1e07b75e22cbaf3dbc2384b9470e372b0549b7b98a peer=1 000294317.44 SetBeatChain: new best=0000000000012378f1e07b7 0000481=2014-06-30 12:27:52 progress=0.999253 2014-06-30 13:17 00058814-06-30 13:17:44 getblocks 94179 to 0 limit 500 peer=138 000178(2552b04655611c92008e0a775b1ef9d53c4329154ab4 peer=13 00012827+44 SetBeatChain: new best=0000000000048425bf2dcba 0001176014-06-30 12:30:11 progress=0.99933 2014-06-30 13:17:44 0001382-06-30 13:17:44 getblocks 94191 to 0 limit 500 peer=138 0001470 94192 to 0 limit 500 peer=140 2014-06-30 13:17:44 rece 00016126-530 13:17:44 getblocks 94191 to 0 limit 500 peer=138 0001470 94192 to 0 limit 500 peer=140 2014-06-30 13:17:44 rece 00016126-530 13:17:44 getblocks stopping at limit 9583 0000 00020594181 to 0 limit 500 peer=137 2014-06-30 13:17:44 getblock 00027930.999936 2014-06-30 13:17:44 getblocks 94191 to 0 limit 500 0002499ed5520b0c2d311c0744 peer=137 2014-06-30 13:17:44 getblock 00037930.999936 2014-06-30 13:17:44 ProcessBlock: ACCEPTED 20 00027930.999936 2014-06-30 13:17:44 ProcessBlock: ACCEPTED 20 00037930.999936 2014-06-30 13:17:44 ProcessBlock: ACCEPTED 20 0003794192 to 0 limit 500 peer=137 2014-06-30 13:17:44 Getblock 00037930.999936 2014-06-30 13:17:44 ProcessBlock: ACCEPTED 20 0003794162 to 0 limit 500 peer=137 2014-06-30 13:17:44 Getblock 0003794092040000000002bf04948431b946158f5251217e9658520b022 00037950.999936 2014-06-30 13:17:44 ProcessBlock: ACCEPTED 20 000374162 to 0 limit 500 peer=137 2014-06-30 13:17:44 Getblock 00037950.999936 2014-06-30 13:17:44 ProcessBlock: ACCEPTED 20 00037450.999936 2014-06-30 13:17:44 ProcessBlock: ACCEPTED 20 00037450.999936 2014-06-30 13:17:44 ProcessBlock: ACCEPTED 20 00037450.999936 2014-06-30 13:17:44 ProcessBlock: ACCEPTED 20 00033810000000002bf049484b94615852511676853548bc1440 0033810000000002bf0496484b94615852511676855211676	75e222 :44 Pr 7 201 tblock 2014 455611 4 Proc 2014- ived b n data =58.80 to 0 1 000000 cocks 9 peer= ting 1 d311c0 14-06- ks 941 eer=14 mg 1 c <u>663043</u>	cbaf3dc2384b94270e ocessBlock: ACCEPT 4-06-30 13:17:44 get 06-30 13:17:44 con cs2008e0a775b1ef3d essBlock: ACCEPTED 06-30 13:17:44 con lock 0000000001da base 2014-06-30 1256 tx=411917 da imit 500 peex=135 4795b5e74a43a43d0ch 4192 to 0 limit 500 140 2014-06-30 13: changed transactic 744 height=94217 30 13:17:44 getbloc 93 to 0 limit 500 93 to 0 limit 500 b height=94218	372b0549b7b958 height D 2014-06-30 13:17:44 tblocks 94190 to 0 lim 500 peer=140 2014-06- mitting 6 changed tran 3c4329154b4b height=9 2014-06-30 13:17:44 g locks 94187 to 0 limit ea0ccd4062f90aff3b22b 13:17:44 SetBestChain t=2014-06-30 13:17:44 ge 8ce9ea917e6f34c4b00b7f peer=138 2014-06-30 17:44 received block 000 to coin database c24 vork=5.801388 tz=	=94214 getblc it 500 30 13:1 saction 4215 1 etblock 500 pp d09f127 : new b progre tblocks a87c0a1 13:17:4 0000000 2014-06	log2 peers 17:44 15:to log2_w xs 941 eers13 754560 bess=0. s 9333 14d50a 44 get 000047 000045 135 getbl 0002bf 5-30 1	<u>uork=58.801115</u> <u>4188 to 01mit 5</u> <u>4188 to 01mit 5</u> <u>4188 to 01mit 50</u> <u>50 to 01mit 500</u> <u>50 to 01mit 500 <u>50 to 01mit </u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u></u>	<pre>tx=411909 date 00 peer=135 20 3:17:44 getbloc 0000000064844 2014=06=30 13: =411912 date=2 peer=137 2014 17:44 getblocks peer=138 2014- cdd4052f90aff3b 0 13:17:44 Proc peer=136 2014- 7:44 getblocks 1imit 500 peer 9cd251c217ee965 Chain: new best=0 Chain: new best=0 0 the the the the the the the the the the</pre>

Figure 25. The debug.log file contains the same information found in the Litecoin debug log. This includes start and stop application time as well as transactions sent from the wallet. Timestamps are in UTC.

The HKU\[User SID]\Software\DarkCoin registry key appeared to be the only notable change to the

registry during installation.

Oddly enough, the current version of Darkcoin does not contain an uninstaller and was not picked up on as an installed program in the control panel. For this experiment, Darkcoin was not uninstalled to determine what type of data gets left behind.

String searching memory with known user Darkcoin addresses contained references to addresses involved in transactions during the current user session (Fig. 26).

🔍 AccessData Fl	TK Imager 3.1.3.2			• x
<u>File V</u> iew <u>M</u>	ode <u>H</u> elp			
🗛 🔬 🗣 🖴) 🚖 🖾 🖬 🗖	<u>a</u> a =	📴 🔍 🗋 🖹 📾 🐱 📽 💡 ↓	
Evidence Tree		X	cc2a71a0 17 00 01 16 54 68 69 72-64 20 44 61 72 6B 63 6F ···· Third Darkco	
E 64ddf7e.4	1b1-e4fc-86d8-9d2a	9f37fb77.vmem	cc2a71b0 69 6E 20 41 64 64 72 65-73 73 63 6F 28 00 01 04 in Addressco(
	101-6410-0000-0028	0/07/077.VIII6III	cc2a71c0 6E 61 6D 65 22 58 65 46-42 4A 55 34 32 34 68 6F name"XeFBJU424ho	
			cc2a71d0 76 56 35 68 66 4E 35 52-55 65 48 36 41 46 54 74 VV5hfhSRUeH6AFTt	
			cc2a71e0 73 31 47 58 73 31 42 70-18 00 01 17 53 65 63 6F s1GXs1Bp ···· Seco	
			cc2a71f0 6E 64 20 44 61 72 6B 63-6F 69 6E 20 41 64 64 72 nd Darkcoin Addr	
			cc2a7200 65 73 73 73 28 00 01 04-6E 61 6D 65 22 58 66 63 esss(name"Xfc	
			cc2a7210 36 61 33 57 42 35 73 62-34 5A 64 70 67 39 67 6D <mark>6a3WB5sb4Zdpg9gm</mark>	
			cc2a7220 31 6B 79 74 36 39 41 69-32 44 56 5A 33 6D 48 39 <mark>lkyt69Ai2DVZ3mH9</mark>	
			cc2a7230 12 00 01 11 46 69 72 73-74 20 74 72 61 64 65 20 ····First trade	
			cc2a7240 74 6F 20 41 4A 80 2F D6-28 00 01 04 6E 61 6D 65 <mark>to AJ·/Ö(···name</mark>	
			cc2a7250 22 58 6D 55 4A 43 55 56-50 4C 54 4D 63 46 79 4A "XmUJCUVPLIMcFyJ	
Hex Value Interpre	eter	×	cc2a7260 31 76 46 39 53 70 57 4D-41 6D 43 53 77 69 75 72 1vF9SpWMAmCSwiur	
Туре	Size Value		cc2a7270 78 57 63 00 16 00 01 15-46 69 72 73 74 20 74 72 xwcFirst tr	
	1.8	î	cc2a7280 61 64 65 20 74 6F 20 55-62 75 6E 74 75 01 00 A3 add to Ubuntu - 8	
signed integer unsigned integer	1.8	=	cc2a7290 28 00 01 04 6E 61 6D 65-22 58 68 6D 72 4D 6E 65 (···name"XhmrMne cc2a72a0 39 66 56 33 46 72 79 44-57 77 4D 73 54 53 64 34 9£V3FrvDWwMaTSd4	
FILETIME (UTC)	8		$Cc2a72b0$ S^{2} S	
FILETIME (local)	8		$\frac{1}{2} \frac{1}{2} \frac{1}$	
DOS date	2	-	cc2a72d0 4B 48 6B 10 9C 93 67 4A-EE F0 E7 D9 1E 47 65 F7 KHk+dJi6d0-Ge+	
•		•	cczarzeo 46 76 7C A1 71 1C 5E 68-CE 47 51 CD 46 00 000 FV jq ~hig0rt0	
	0.1.W. F	0.0	cc2a7250 0D 00 01 04 70 6F 6F 6C-35 00 00 00 00 00 00 00 00 00 00 00 00 00	
Byte order:	Little endian	C Big endian	cc2a7300 2E 00 01 E1 63 01 00 A3-67 B1 53 00 00 00 02 1śc£g±5!	-
Properties	Hex Value Int	Custom Conte	Cursor pos = 3425333872; log sec = 6690105	
564ddf7e-41b1-6	e4fc-86d8-9d2a9f	37fb77.vmem	NUM	1 //

Figure 26. String searching addresses in memory contains the same type of data as stored in the wallet.dat file. However, this can also be used to determine some transactions that have occurred in this user session.

Darkcoin operates on port 9999 and like Litecoin runs off of the Bitcoin protocol. However, it does

appear to contain some unique message types. Darkcoin's network traffic retains the typical inv,

getdata, and tx structure indicating a transaction and can be manually analyzed in the same method

described before (Fig. 27). The dseep message appears to be a unique message type to Darkcoin.

However, according to a developer post, this message pertains to masternode ping messages and is not

related to a particular transaction ¹⁷. The network ID for Darkcoin is commonly an 'X'.

¹⁷ Duffield, Evan. <u>New version 9.4.0 and 10.8.0</u>. 5 May 2014. <https://darkcointalk.org/threads/new-version-9-4-0-and-10-8-0.690/>.

File Edit View Go Capture Analyze Statistics Telephony Iools Internals Help • • • • • • • • • • • • • • •
Filter tc.port == 9999 Expression Clear Apply Save No. Time Source Destination Protocol Length Info 994 14:57:09.497467 192.168.139.148 54.254.230.222 TCP 305 49165 > distinct [PSH, ACK] seq=2340 Ack=3637 win=63619 Len=251 Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source Image: Source
No. Time Source Destination Protocol Length Info 994 14:57:09.497467 192.168.139.148 54.254.230.222 TCP 305 49165 > distinct [PsH, ACK] Seq=2340 Ack=3637 win=63619 Len=251 * m * * * * * * B Frame 994: 305 bytes on wire (2440 bits), 305 bytes captured (2440 bits) * * * * B Ethernet II, Src: Ymware_37:fb:77, (00:0c:29:37:fb:77), DSt: Ymware_f6:cf:90 (00:50:56:f6:cf:90) * * * B Transmission Control Protocol, Src Port: 49165 (49165), DSt Port: distinct (9999), Seq: 2340, Ack: 3637, Len: 251 * * Data (251 bytes) * * * * * 0030 f8 83 6b 2f 00 00 bf 0c 6b d7 78 00 00 00 00 , k.tx, * * * 0050 00 00 00 00 00 00 00 00 00 00 c1 c2 74 3a d1 4e 4d .k.9:, bt:.M * * * 0060 1b 6b f9 39 3a ff a8 9e 0c c1 62 74 3a d1 4e 4d .k.9:, bt:.M * * * 0060 00 64 00 00 00 00 00 00 9f 09 * * * * *
994 14:57:09.497467 192.168.139.148 54.254.230.222 TCP 305 49165 > distinct [P5H, ACK] Seq=2340 Ack=3637 Win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Len=251 Image: Seq=2340 ack=3637 win=63619 Bits Image: Seq=2340 ack=
<pre></pre>
 B Ethernet II, src: Vmware_37:fb:77 (00:0c:29:37:fb:77), Dst: Vmware_f6:cf:90 (00:50:56:f6:cf:90) B Internet Protocol Version 4, src: 192.168.139.148 (192.168.139.148), Dst: 54.254.230.222 (54.254.230.222) B Transmission Control Protocol, Src Port: 49165 (49165), Dst Port: distinct (9999), Seq: 2340, Ack: 3637, Len: 251 B Data (251 bytes) 0030 f8 83 6b 2f 00 00 bf 0c 6b bd 74 78 00 00 00k/ k.tx 0030 f8 83 6b 2f 00 00 bf 0c 6b bd 74 78 00 00 00k/ k.tx 0040 00 00 00 00 00 00 00 00 00 bc 77 bc e9 01 00
B Transmission Control Protocol, Src Port: 49165 (49165), Dst Port: distinct (9999), Seq: 2340, Ack: 3637, Len: 251 B Data (251 bytes) 0030 f8 83 6b 2f 00 00 bf 0c 6b bd 74 78 00 00 00 00k/k.tx 0040 00 00 00 00 00 00 e3 00 00 0b 7 bc e9 01 00
0040 00 <
0000 a2 15 06 16 13 27 16 17 98 97 10 <

Figure 27. Darkcoin contains most of the same message types as Bitcoin and Litecoin. It can also be manually analyzed in the same fashion. Wireshark does not recognize Darkcoin and interprets it as TCP traffic running on port 9999.

Darkcoin's claim to fame is the anonymous Darksend feature which allows transactions to be 'completely anonymous'. However, as of Release Candidate 3, there are currently many restrictions on this functionality. Because of this, there are not enough users participating to actually mix the transactions correctly. So while this data does not include transactions using Darksend, it can give an examiner an idea of where to start looking for transactions.

Conclusion

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.