VideoCoin - A Decentralized Video Encoding, Storage, and Content Distribution Network

Devadutta Ghat
devadutta@videocoin.io

November 7, 2017
Version 1.0.30

Abstract

Video constitutes a staggering amount of internet data and by 2021, it is predicted to form 82% of all Internet traffic [1].Video has historically been a driving force for innovation on the Internet, just like Bitcoin and Ethereum have been for decentralized networks.VideoCoin brings these two revolutionary ideas together by building a decentralized video infrastructure that will power the next generation of video applications. VideoCoin is a decentralized network that provides cloud video infrastructure like video Encoding, Storage and CDN in the form of a peer-to-peer algorithmic market. VideoCoin runs on a new blockchain with a native protocol token where miners earn VideoCoins by providing video infrastructure services and clients spend VideoCoins to rent these services.

Note: VideoCoin is undergoing vigorous research and development and content in this paper can change on a regular basis. Refer to https://videocoin.io/ for the latest version of this paper.

Contents

List of Figures

Definitions

Clients	Consumers of the VideoCoin Network infrastructure
Miners	Producers of compute, bandwidth and storage capacity
Nodes	Each instance of VideoCoin miner in the network. Can be physical machine or virtual
Frame	Electronically coded still image in a video
σ	Amount of VideoCoins staked to the network by the account
Ψ	**Barrier to Entry** - Minimum number of VideoCoins to be staked in order to qualify for participating in the verifier pool.
ω	Watt - A unit of resources on the VideoCoin Network. Weighted average of CPU Time, GPU Time, Storage and Internet Bandwidth
π	Price per Watt in VideoCoins
μ_{pay}	Micro-transaction amount in VideoCoins
B_{reward}	Block Reward
τ	**Clock Tick** - New event in the network event loop
W	**Confirmation Window Size** - Number of confirmed blocks for which σ transactions have to wait.
κ	**Clout** - Minimum σ balance over W new blocks
Θ	Total VideoCoins staked in the network by all accounts
GOP	Group Of Pictures in a video

1 Introduction

Modern cloud video infrastructure that powers a variety of video applications on the internet – like YouTube, Netflix, Hulu, Twitch and Amazon Video – consists of a common set of technologies that enable video streaming. We describe the basic building blocks below

1.1 Video Infrastructure

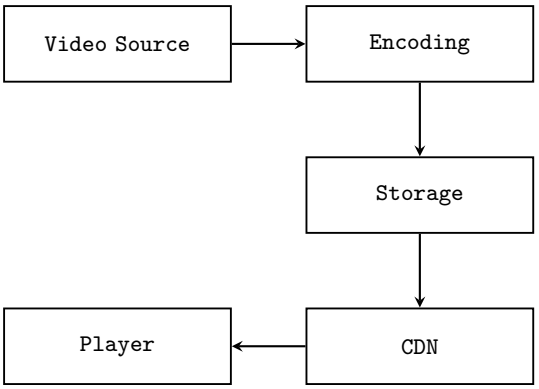

Figure 1: Video Infrastructure

Cloud Video Infrastructure has three core components:

- **Encode** - *Creating video bitstreams that match requirements of a client* - Typically a video is consumed on different devices and each device has its own codec, colorspace and resolution requirements. Encoding and transcoding makes playback on different devices possible by taking a source video and transcoding it to match device requirements. Modern video infrastructure makes use of codecs like H265, H264, VP9 and VP8 to achieve high degrees of compression, but it comes at the cost of high CPU requirements [3].
- **Store** - *Storing Video in different formats, bitrates and codecs post encoding* - With increased resolution Video files are occupying more storage space than ever before and 4K videos are now the preferred format for consumption for high-end devices. End-consumers view videos in an ever increasing number of devices and encoding for each device just in time is compute intensive. Thus, videos are typically encoded once and stored in a storage network for later consumption.

- **Deliver** - *Delivering chunks of video over the Internet using servers that are geographically in close proximity to consumer* - To deliver a good user experience by avoiding long buffering times, videos are generally cached in a location geographically close to the end-consumer. Modern video infrastructure uses content delivery networks to provide with this function.

2 VideoCoin

VideoCoin implements decentralized versions of core video infrastructure components and creates a new class of video miners who compete with other miners to earn rewards by providing CPU and/or GPU cycles for video encoding, disk space for video storage and Internet bandwidth for CDN services. These video miners can run on data center servers, household desktop computers or even mobile phones, creating a powerful network of computers which would otherwise go unused. In fact, over 30% of Data Center servers are comatose [2]. VideoCoin incentivizes miners to use such unused resources and provide as much compute, storage and bandwidth available.

Main components that powers the VideoCoin network are

2.1 Distributed Video Encoding

Large scale video processing is a complex distributed systems problem. Encoding videos to match different devices, resolutions, and codecs consumes a lot of compute resources. While Moore's law has been driving prices of individual CPUs lower by the day, compute capacity of CPUs have not caught up to provide enough power to process videos. In our studies (see Section 12), even high end cloud CPU instances performed poorly, at 1-2 FPS for transcoding a 4kx4k video VR video. We need at least 30 FPS of processing power to deliver real-time streaming of videos and the most scalable and cost effective way of achieving this is by using a distributed encoder. Advancements in GPU technologies and fixed function video encoders have alleviated this problem to a certain extent, but these fixed function encoders have limited encoder configurations available [4] and is significantly more expensive than renting CPUs. VideoCoin addresses this problem by implementing a Distributed Video Processing platform, which splits a video encoding task into several sub tasks and processes them in parallel, thus effectively producing much higher frame rates on commodity hardware. Encoders are run on the open source media framework, ffmpeg [5] inside a secure container, so rogue applications cannot damage the host computer.

Figure 2 shows how VideoCoin's Distributed encoder functions

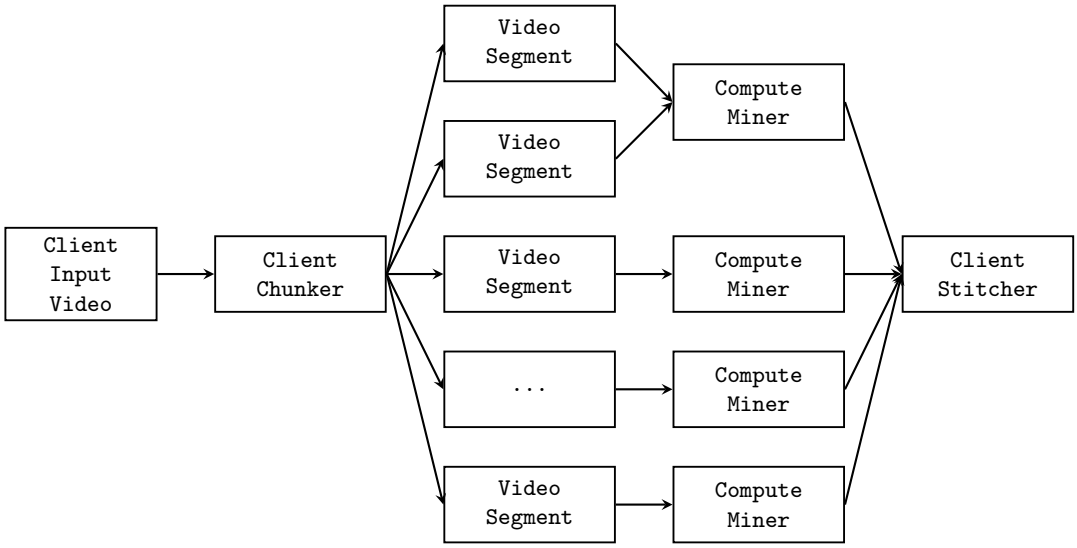

Figure 2: Distributed Video Encoding

1. Source video is input to the VideoCoin Client Software, along with encoder configuration parameters, like resolution, bitrate, and codec.
2. Client software splits input video into segments
3. Client polls available compute miners to accept current transcoding job
4. Compute miners accept the job and client issues transactions
5. Miners finish the job, provide proof of work
6. Client makes payments to miners

2.2 Distributed Video Storage Network

VideoCoin implements a distributed file system, called Sonata, on top of commodity hardware provided by storage miners. Sonata is a scalable, fault-tolerant, distributed storage system that works closely with a wide variety of concurrent data access applications. Videos are traditionally stored as large monolithic files making them susceptible to failure and hard to transport. To remedy this, Sonata splits videos into playable segments and distributes them across the storage network created on top of disk space provided by storage miners. Sonata is inspired by modern advancements in distributed storage including the highly scalable HDFS [6].

2.3 Content Distribution Network

A CDN is a geographically distributed network of proxy servers and their data centers [7]. In VideoCoin, distribution miners act as paid proxy servers for storage miners who store data. Decentralized networks by nature are geographically distributed as miners across the globe participate in the network. However, not all miners have access to the same Internet bandwidth which is essential for good quality video streaming. VideoCoin builds an algorithmic market for clients to negotiate contracts directly with distribution miners and storage miners to deliver high quality video to the end-user via a proxy server that is in close geographical proximity.

3 Accounts

VideoCoin accounts are objects with a 20-Byte address which have the following fields

- `Type`
  - `DC (Debit/Credit) Account - Controlled by Private Keys`
  - `Contract Account - Controlled by contract code`
- `Nonce`
  - A Counter used to make sure each transaction is executed only once
- `Balance`
  - Current Spendable Balance of VideoCoins
- `Stake(`σ`)`
  - Current VideoCoin Balance Staked to the Network
- `Contract Code`
  - Video Contract
  - Smart Contract
- `Storage`
  - Storage space for the account

Like any other smart contract protocol, “Contracts” are pieces of code that run when initiated by a message.

4 Transactions

Transaction in VideoCoin is signed data structure that stores a message to be sent from one account address to another.

Transactions contain the following data fields:

- `Type`
  - σ`DC Transaction`
  - `DC Transaction`
  - `CR (Challenge/Response) Transaction`
  - `Trigger`
- `Recipient Address`
- `Sender Signature`
- `Amount`
  - Number of VideoCoins to be transferred
- `Parameters`
  - Data Parameters for the Contract Code
- `Maximum Wattage (`ω`)`
  - $\omega_{\max}$ (compute + bandwidth + storage) Maximum power sender is willing to accept for processing this transaction
- Price (π)
  - Price in VideoCoins/Watt sender is willing to pay for compute and bandwidth on the VideoCoin network depending on the type of transaction
- `Reward` (μ_{pay})
  - Micro-payment/Fees the sender is willing to pay for successful confirmation of a CR transaction

π will be set by market demand. π and ω is similar to GAS prices in the Ethereum [8] network and are required to protect the network against DDoS attacks.

5 Blockchain

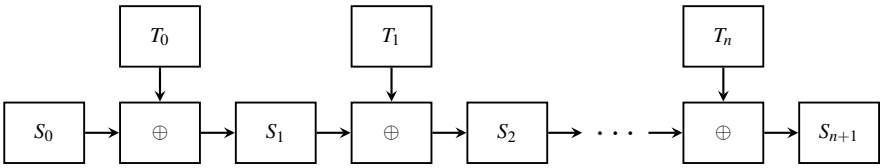

Figure 3: BlockChain and Network Consensus

The VideoCoin blockchain is similar to Bitcoin and Ethereum blockchains and differ mainly in the way block rewards and fees are assigned. A block reward B_{reward} is always assigned to the α node in the verifier pool that created the new block. Fees in the form of π and ω are also assigned to the α node in the verifier pool. Micro-payments μ_{pay} from CR transactions are always earned by miners. Block rewards and ω are always a fraction of the mining costs, so the highest incentive on the network is for miners to rent out storage, compute and bandwidth.

6 Network Consensus

Network Consensus in VideoCoin is maintained by a pool of **Verifier Nodes** and useful proof of work for miners. At every clock tick τ, a pool of verifiers are picked using a pseudo random process where the probability of being a part of the verifier pool is directly proportional to the number of coins staked into the network.

6.1 Proof of Stake

In order to enter a verifier pool, any node in the VideoCoin network can stake tokens into an unspendable balance on the account called the stake balance [see Section 3]. Transactions to increase or decrease stake balance can happen only when the sender sends tokens to itself and setting transaction type to increase account stake(σDC). In order to protect the network from rogue nodes that just transfer balance into a stake account to win block reward and immediately move the balance to another spendable account, the network does the following

1. Transactions from DC Balance to Stake Balance are not confirmed for Confirmation Window W with n blocks
2. Maintain a barrier to entry (Ψ), which is defined as the minimum Stake Balance (σ) an account has to hold in order to qualify for the Verifier Pool

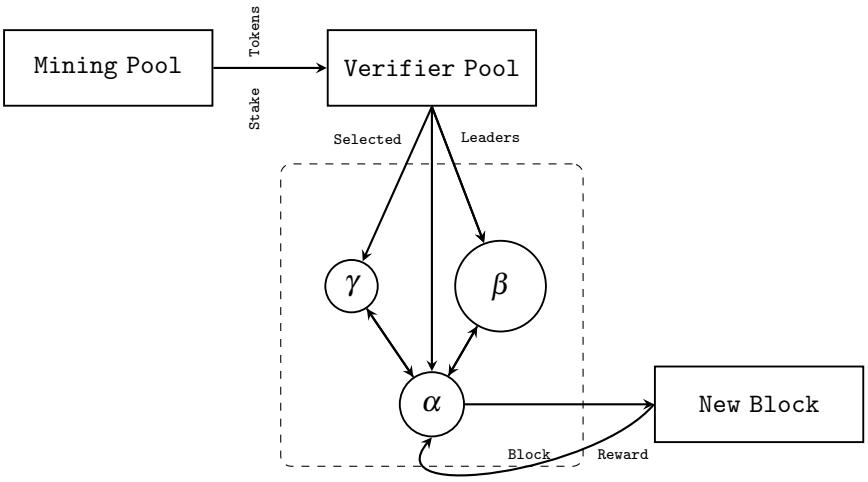

Figure 4: Verifier Pool

3. Probability of winning a slot (ρ_v) in the verifier pool and gaining α voting rights is proportional to the clout (κ) of the account, which is defined as the minimum value of staked balance σ over a time window W Mathematically,

$$\kappa = \begin{cases} \min(\sigma_{0\ldots n}) & \text{if } \sigma > \Psi \\ 0 & \text{if } \sigma \leq \Psi \end{cases} \tag{1}$$

Where, σ_i is the the staked balance of the account in question at block i of the Confirmation Window W.The probability of winning a slot in the pool (ρ_v) is calculated as below

$$\rho_v = \begin{cases} \frac{\kappa}{\Theta} & \text{if } \kappa \geq \sigma \\ 0 & \text{if } \kappa < \sigma \end{cases} \tag{2}$$

Where, Θ is the total number of coins staked by all nodes in the network.

As illustrated by Figure 4, at every clock tick τ, a node in the VideoCoin network does the following

1. Choose set of addresses that are part of the verifier pool, based on a seed derived from a block confirmed W blocks ago.

2. Randomly shuffle the pool of addresses (based on the same seed) and pick top three addresses as α, β and γ

3. Check if self is α, if yes, verify the block, write it to the blockchain

4. Earn block reward

5. All other nodes in the verifier pool verify new block written by α and if the block is valid, verifier pool is dissolved

6. If the majority of the verifier pool determine the block is invalid, the entire amount of VideoCoins staked in the network by α will be transferred from its σ balance and burned.

7. Beta will become the new Alpha and steps 3-7 will repeat

In this model, block rewards are earned by nodes in the verifier pool and transaction fees are earned by the miners.

Note: *Proof-of-stake algorithms are under intense development and we expect to achieve a practical implementation in stages. Refer to our upcoming research and technical reports on our website.*

6.2 Qualifying for the Verifier pool

In order to incentivize mining nodes to enter the verifier pool, difficulty of the proof of work needed to prove video ownership is inversely proportional to the VideoCoin staked into the network.

$$\omega_{expected} \propto \frac{1}{\sigma} \tag{3}$$

Miners can choose to automatically move a percentage of the transaction reward to an unspendable wallet account, or even deposit VideoCoins into the account to earn higher probability of getting elected into the verifier pool. Funds moved into the stakeholder account cannot be moved out to a spendable account until n blocks, defined by time window W are fully confirmed on the blockchain.

7 Mining

Miners earn VideoCoins by renting out storage, compute and bandwidth available on their systems. Each mining operation is supported by a corresponding proof.

7.1 Storage Mining

Videos on the VideoCoin network are stored as fault tolerant, playable chunks. Videos stored in standard container formats, which include audio are first split into video segments based on file size by the client redundancy factor. The main difference between splitting a binary file at arbitrary intervals and chunking a video file, is that in case of video segmenting, each segment is by itself fully playable. This is a very important characteristic of the storage network which is used by distribution miners to provide a playable video even in case parts of the video are missing. The client then generates a set of random challenge salts, S, where $|S|$ depends on amount of time the video has to be stored and video size. The client encoder then chunks videos into playable segments and constructs a Merkle Tree for every segment in the video. These hash digests are requested for proof of storage.

7.1.1 Proof of Retrievability

Storage miners provide proof of retrievability to prove that they are holding a video segment. Proof of retrievability is performed in the following steps

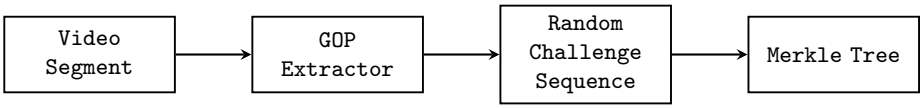

Figure 5: Distributed Video Storage Network

Client

1. Split an input video file into segments $V = \{v_0, v_1, \ldots, v_n\}$
2. Create an array S of challenge sequences $S = \{s_0, s_1, \ldots, s_n\}$
3. Extract $|S|$ Group Of Pictures (GOP) sequences from the video
4. Append challenge sequences to $|S|$ GOP chunks
5. Calculate the Merkle Tree for each of the challenge sequences

The client then stores the challenge sequences, Merkle Root and depth for each challenge sequence and periodically issues a CR message to the miner. The miner then computes the Merkle Proof for the challenge sequence in the CR message and submits proof to the network. Upon receipt of successful proof, the client makes a payment of μ_{pay} to the miner.

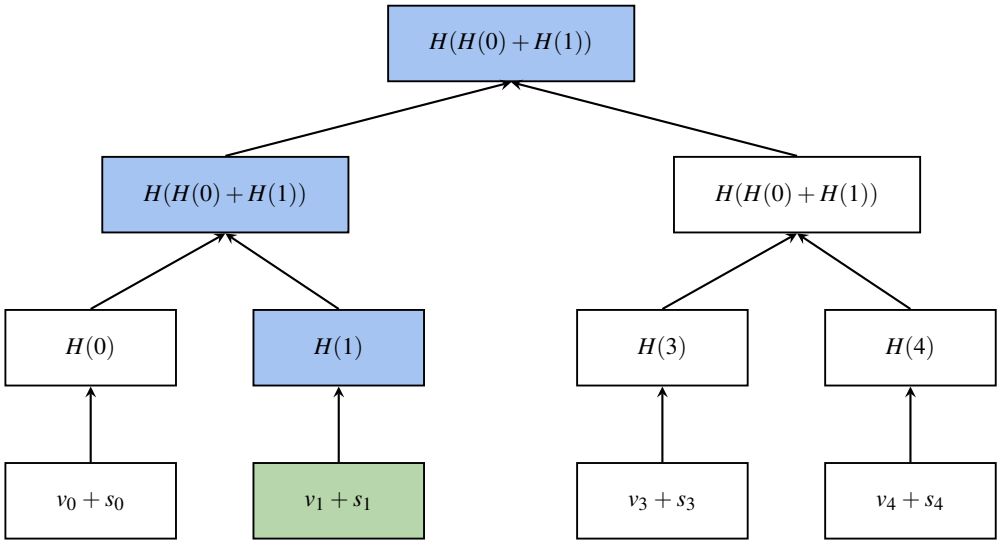

Figure 6: Proof of Retrievability

Note: CR messages can also be issued by a smart contract, thus allowing the client to upload and vanish.

7.1.2 Fault Tolerance

Fault tolerance is achieved by using Reed-Solomon Erasure coding [9], where Erasure coding algorithms break a file into k blocks, and programmatically create m parity blocks, giving a total of k + m = n blocks. Any k of these n blocks can be used to rebuild the video segment. Fault tolerance parameters can be customized by the client while making a storage request.

7.1.3 Other proofs of storage

We are considering the use perceptual hash functions instead of Cryptographic hash functions for establishing whether a video segment is similar to the chunk sent by the miner to prove ownership of the video.

7.2 Relay Mining

Modern cloud storage products like products like AWS S3, Google Cloud Storage and Backblaze provide highly reliable, but centralized storage. Relay mining al-

lows miners to transparently outsource storage to a third party cloud storage. This provides an arbitrage opportunity for miners who have access to surplus cloud storage. Relay mining software for the VideoCoin network is functionally similar to Storage Mining software, with the file system interface replaced by a cloud storage API. This implementation is similar to HDFS [6] and S3A [10].

7.3 Compute Mining

Compute miners execute video functions like transcoding. While any node can participate in the network as storage miners, only nodes that have staked tokens greater than barrier to entry, Ψ, are eligible to perform compute mining operations. General purpose compute in VideoCoin network is managed by smart contracts which are executed by all nodes in the network.

Compute miners prove their output using a novel proof of work called Proof of Transcoding.

7.3.1 Proof of Transcoding

Any modern video codec is (broadly) organized into a hierarchical data structure consisting of Frames $>$ Slices $>$ Macroblocks. Videos codecs take advantage of temporal redundancy and use information in previous or future frames (I,P and B Frames) to compress the current frames. This means that in order to decode a macroblock in a particular frame k, all frames that k depends on have to be decoded first. Proof of Transcoding uses this dependency to create a set of challenges that is used by the client to establish if a transcoding job has been completed per contract. For every new transcode job that needs to be submitted to the network, the client does the following

1. Set up input video and transcode parameters
2. Split the input video into chunks and performs a PUT operation to store the video on the VideoCoin network (if not already stored)
3. For each of the video segments the client needs to transcode
   (a) Extracts a random set of challenge sequence of frames, also known as GOP
   (b) Client performs transcode to match input parameters on these sub-segments and hold the hash value along with a challenge salt either in local storage or a part of another smart contract

4. Client checks the transcode proof and provides additional micropayment to the miner

In order to avoid partial compute attacks, the following protocol is followed At the beginning of every time span, a verifier pool is elected as described in the section 6.1 above.

1. Once a compute miner broadcasts a proof, a verifier node from the pool will do the following
   (a) PULL original source Video from the storage miner
   (b) Verify hash of the Video to confirm source is intact
   (c) Extract a random GOP from the Video, perform the exact transcode operation in the video contract
   (d) PULL the transcoded Video from the compute miner, extract the GOP from the transcoded video
   (e) Check if the two GOPs are exactly the same
2. If the Proof of Transcode is malicious, the node will lose all tokens staked to the network

Both clients and verifiers can confirm work by a compute miner for a very small fraction of the compute cost by just checking transcode on a randomly selected GOP from the video.

7.4 Distribution Mining

Distribution miners primary functions is to act as gateway between the VideoCoin network and Internet. These miners are similar to edge nodes on a CDN [7] and the storage miners are similar to origin servers. Just like compute miners, distribution miners also need to stake more tokens than barrier to enter the verifier pool to be able to perform distribution mining.

Distribution miners establish P2P connections with clients that consume video from the network and act as an intermediary.

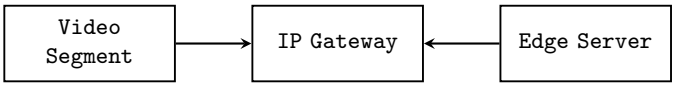

Figure 7: Content Distribution Network

1. Client sends a message of type DISTRIBUTE to nodes in the distribution pool along with parameters to determine video size and minimum bandwidth required for transmission
2. Nodes in the distribution pool willing to accept MicroPay fees for this distribution respond with an OFFER message to the Client
3. The client can then reply with an ACCEPT message to multiple nodes in the distribution pool based on the distribution requirement
4. The distribution node then establishes a P2P connection with the consumption node and delivers the video bitstream

Most of the negotiation during distribution operation happens outside of the blockchain.

7.5 Sandboxing and Isolation

All VideoCoin mining software runs inside an isolated Docker container. Video contract code is designed to run on FFMPEG binaries inside the Docker container and smart contract code is compiled down to binary executable inside the container. This allows for smart contract programming in different languages.

8 Smart Contracts

Contract Chains

Contracts in VideoCoin can be chained with other contracts using `Trigger` messages, allowing complex video operations, an example is shown in Figure 8. In case of chained contracts, certain contracts cannot execute if the output of the previous contract in the chain is not available. Thus, we allow contracts to be tagged dependent and independent. A miner choosing to execute a chain, will execute the entire chain to earn mining rewards. In Figure 8, an input video is transcoded from H264 to H265 while the accompanying audio on track is transcoded from MP3 to AAC and finally multiplexed back into an MP4 file. A contract like this would execute entirely on the same miner node and the contract would be enforced by the verifier pool using proof of transcoding.

9 Scalability

One major concern with all blockchain based decentralized networks is scalability and VideoCoin network is not immune to these problems. Some steps have been

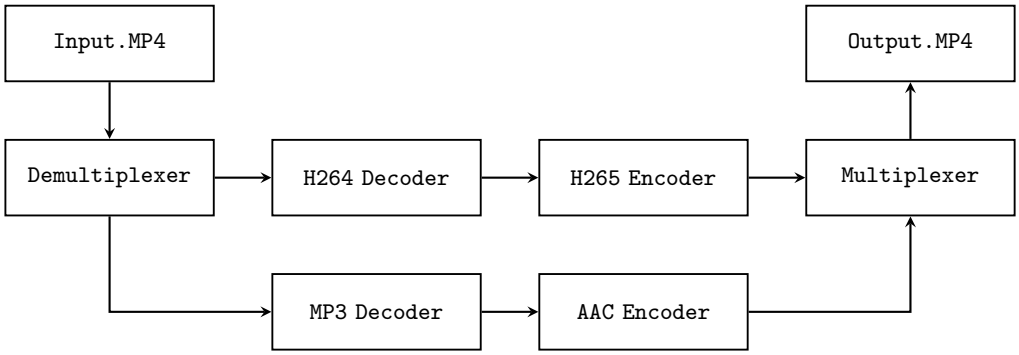

Figure 8: Contract Chains

taken to mitigate these problems. Unlike Bitcoin [11] and Ethereum [8] all transactions and all general purpose smart contract code do not have to be executed by every single node on the VideoCoin network. The verifier pool takes a leading role in maintaining network consensus and other nodes can follow. Miner nodes perform majority negotiations outside of the blockchain through direct messages. Only micro-payments and proofs are verified by the verifier pool when confirming transactions.

10 Applications

VideoCoin is built for applications and several popular video products from the Internet can be reimagined as a decentralized, P2P application. Few examples

- **YouTube Contract:** A client can use storage, encoding and distribution functions provided by VideoCoin to build a decentralized version of YouTube where anyone can upload and view videos and creators can earn VideoCoins directly from the consumer
- **Hulu/Netflix Contract:** Building on top of the YouTube contract, a client can use general purpose smart contracts to add a subscription based payments

11 Attacks

Like any distributed system, a variety of attack vectors exist. Some of these are listed here along with strategies to mitigate the dangers. A detailed technical report

covering security risks and strategies is a part of our future work roadmap.

11.1 Sybil

Sybil attacks [12] involve creating a large number of nodes in the network in the hope of disrupting the network for the attacker's advantage. Since nodes in the verifier pool are selected at random based on the amount of tokens staked, for an attacker to control 51% of the VideoCoin network involves insurmountable financial and computational hurdles and high risk of losing all of the staked tokens if the verifier pool deems the alpha node rogue. Not only does an attacker have to control 51% of all nodes in the network, but the attacker also needs to end up with a verifier pool with 51% of nodes that vote in his favor, making the attack financially unfeasible.

11.2 Illegal Content

A common problem with video distribution networks is piracy and illegal content. These problems can be addressed with a client reputation system and a smart contract based content flagging system. Several reputation systems are being evaluated at VideoCoin, findings will be published in an upcoming technical report.

12 Future Work

VideoCoin is a work in progress and a detailed roadmap of our technical research can be found at https://videocoin.io. The following research activities are currently being undertaken

- **Security:** Investigation into different attack vectors against VideoCoin and mitigation strategies
- **Proof of Stake:** Reference implementation of VideoCoin's proof of stake model
- **Performance analysis of Proof of Transcode:** Establishing difficulty of proof of transcode will require emperical analysis of various systems miners plan to execute VideoCoin software on

References

[1] Cisco. *Cisco Visual Networking Index: Forecast and Methodology, 2016 – 2021*. 2017.

[2] Jonathan Koomey and Jon Taylor. *Zombie/Comatose Servers Redux*. 2017.

[3] Ana Rodrigues. *H.264 vs H.265 - A technical comparison. When will H.265 dominate the market?* `https://goo.gl/Ft1h8V` 2016.

[4] Nvidia. *NVENC Support Matrix*. `https://goo.gl/xX5KWX`.

[5] FFMPEG. *A complete, cross-platform solution to record, convert and stream audio and video*. `https://www.ffmpeg.org/`.

[6] Apache Hadoop. *HDFS*. `https://hadoop.apache.org/`.

[7] Wikipedia. *CDN*. `https://goo.gl/SWcn6k`.

[8] Ethereum. *A Next-Generation Smart Contract and Decentralized Application Platform*. https://github.com/ethereum/wiki/wiki/White-Paper.

[9] Wikipedia. *Erasure code*. `https://goo.gl/DuJzCV`

[10] Apache Hadoop. *S3 Support in Apache Hadoop*. https://wiki.apache.org/hadoop/AmazonS3.

[11] Satoshi Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. 2008.

[12] Wikipedia. *Sybil attack*. `https://goo.gl/E9bV5c`.

[13] Protocol Labs. *Filecoin: A Decentralized Storage Network*. 2017.

[14] Shawn Wilkinson et al. *Storj: A Peer-to-Peer Cloud Storage Network*. 2016.

[15] Nick Szabo *Smart Contracts: Building Blocks for Digital Markets*. 1996.

Disclaimer

This white paper is for information purposes only. The VideoCoin Development Association (VDA) does not guarantee the accuracy of or the conclusions reached in this white paper, and this white paper is provided as is. VDA does not make and expressly disclaims all representations and warranties, express, implied, statutory or otherwise, whatsoever, including, but not limited to: (i) warranties of merchantability, fitness for a particular purpose, suitability, usage, title or non-infringement; (ii) that the contents of this white paper are free from error; and (iii) that such contents will not infringe third-party rights. VDA and its affiliates shall have no liability for damages of any kind arising out of the use, reference to, or reliance on this white paper or any of the content contained herein, even if advised of the possibility of such damages. In no event will VDA or its affiliates be liable to any person or entity for any damages, losses, liabilities, costs or expenses of any kind, whether direct or indirect, consequential, compensatory, incidental, actual, exemplary, punitive or special for the use of, reference to, or reliance on this white paper or any of the content contained herein, including, without limitation, any loss of business, revenues, profits, data, use, goodwill or other intangible losses.

