THE HYPERSPHERE & FINTECH

A Whitepaper on Financial Use Cases– *Ensuring Integrity in Online and POS Transactions*

Richard K Williams and Evgen Verzun

VALUE PROPOSITION

Overcoming the intrinsic security vulnerabilities of the Internet, TCP/IP, and SSL/TLS, the HyperSphere and its enabling technology, the Secure Dynamic Network & Protocol (SDNP), represent the world's only fully-decentralized cybersecure communication and privacy network, uniquely capable of protecting credit card, POS, and bank transactions against fraud, theft, and a broad range of cyber-attacks. As a privacy network, the HyperSphere uniquely protects personal information, online identity, blockchain and databases from unauthorized access, imposter fraud, and theft. The projected economic benefit of the HyperSphere to the financial services sector in combatting credit card fraud and account hacking is incalculable, making the HyperSphere indispensable fintech for the financial sector and e-commerce.

SUMMARY

The HyperSphere is an open-source fully-decentralized autonomous distributed cloud and e-commerce platform comprising a meshed network of installed application software nodes (HyperNodes) able to perform cloud-based communication, computing, and disaggregated data storage on any networked device (including private and public servers as well as on PCs and mobile devices). The HyperSphere supports cloud-connected devices (IoT, V2V, V2X) and all forms of e-commerce and banking.

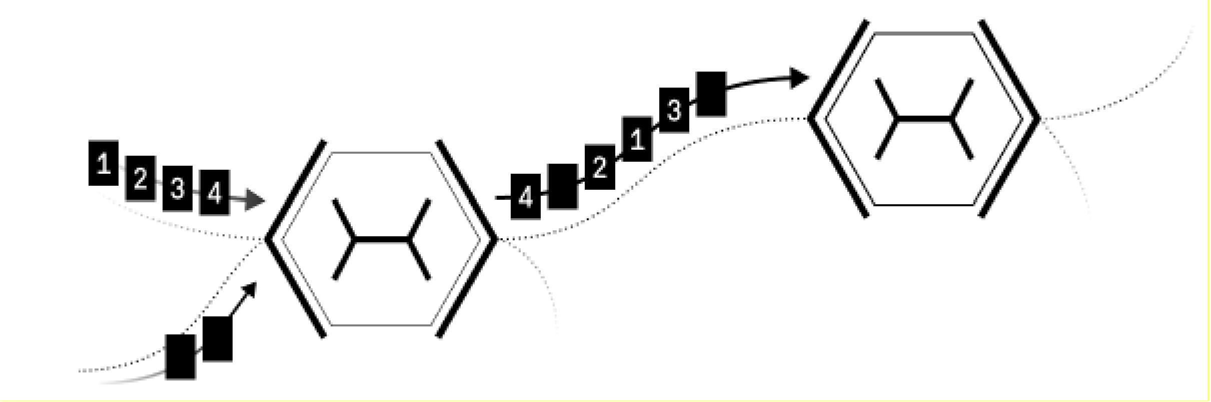
At the heart of the HyperSphere is an innovative (and patented) communication protocol– the Secure Dynamic Network & Protocol (SDNP). SDNP technology protects transactions and communications against a full spectrum of network attacks (including packet hijacking, address spoofing, and man-in-the-middle attacks), while confounding metadata collection and traffic surveillance (methods commonly used in the Internet to execute profiling and identity theft).

First developed for mission critical applications on professional communication radio networks, SDNP's cyberattack resilience has been validated globally for nearly two decades by municipal emergency services (police, ambulance, fire) and by the US military (in FIPS140-2 compliant applications). SDNP network development for commercial applications commenced in 2012 with the world's first SDNP-based hypersecure messenger StealthTalk (available on the Google and Apple app stores). A secure email client StealthMail soon followed. The development of a decentralized version of SDNP (i.e. d'SDNP) ensued in 2014, ultimately giving birth to the HyperSphere as a fully distributed open-source global e-commerce and communication platform.

As a decentralized meshed network, the HyperSphere has no network operator, supervisor, or master key holder. Instead each HyperNode operates autonomously, locally managing packet transport on a hop-by-hop basis without intervention of a superior authority. As such, users don't need to trust that a network operator won't hack their calls or intercept their transactions because there is no network operator to trust. In operation, destination addresses are dynamically supplied to communicating devices and servers by a session dispatcher, a separate group of independently operating HyperNodes involved in routing but with no access to any packet's payload.

HyperNode host devices carrying content have no idea what a SDNP packet is, what kind of data it contains, or where it is going. This decentralized operation is analogous to a shipping company where the Chicago office knows to forward an incoming package to Tokyo but doesn't know what will happen to the package once it gets there. And since no device host is able to monitor what a HyperNode is doing, transactional security does not rely on trusting any cloud, server, PC or mobile device carrying the content. There is no benefit for a cyber-criminal to bribe a corrupt network operator, because the operator has no more information than anyone else – nothing. In this manner, the HyperSphere faithfully delivers trusted transactions even over a trustless network.

Another reason SDNP communication is vastly superior to the Internet's TCP/IP protocol, is because no data packet ever contains sufficient information to be meaningful. In SDNP transport, data is fragmented into snippets of useless data, each sent over different paths through the HyperSphere's meshed network, and reassembled only in targeted edge devices. Meaningful content can only be recovered in the transacting parties' devices participating in a session. Outsiders and interlopers lack the credentials to join the session. SDNP data packets carry no useful metadata either because they only contain single-hop routing instructions, i.e. the IP address of the next HyperNode. Each hop also employs dynamic concealment comprising scrambled, encrypted payloads with no master key exchange.

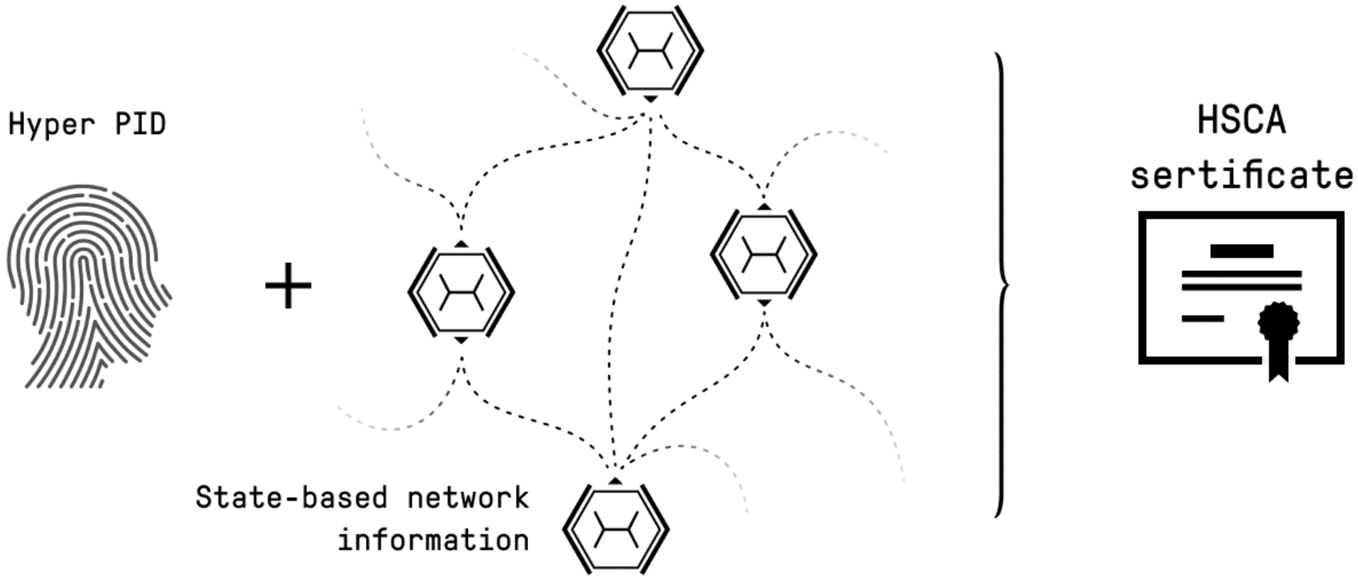


The HyperSphere can be deployed over any and all existing public clouds and private networks (including AWS, Azure, GWS, cellular and cable networks) simply by installing a software application (known as a HyperNode) into a device or a group of devices. Aside from the initial software installation, no support or involvement of a network operator is required to host HyperNodes. HyperNodes can be deployed in public clouds or can be hosted exclusively in privately owned servers. Banking applications using the HyperSphere may execute hypersecure data transport on either of these public or private clouds. The HyperSphere appears as a homologous network of interoperable nodes despite being hosted on disparate clouds and a wide range of device types.

As such the HyperSphere requires no infrastructure or capital investment to deploy, unobtrusively coexisting with TCP/IP based communication. Network operators see HyperSphere traffic but have no idea of what the packets are carrying or where they are going, thereby preventing unauthorized intervention or meaningful monitoring.

For financial services, SDNP communication is enabled through an API link installed into a bank's application software interface. The link operates as a portal to the HyperSphere's meshed network for securely transporting application and client data using SDNP (not by the Internet's TCP/IP). The HyperSphere's routing is transparent to users, requiring no noticeable change in the software's UI/UX (except for offering superior security and privacy benefits) and no learning curve for its clients.

The HyperSphere also offers unique provisions for personal or corporate identity validation, AAA device and user authentication, and blockchain processing for irrevocable incorruptible transactional record keeping. It also supports independent verification of true identity by banks and government agencies. At the heart of this technology is the HyperSphere's network native certificate authority or HSCA. When establishing accounts, HSCA-certificates cryptographically combine hashed personal information (HyperSphere PID) with state-based network information, information that cannot be known by any party except those requesting the HSCA-certificate. Thereafter, personal assets including devices, software installations, files (including documents, record, photos and video), blockchains, wallets, bank accounts, stocks, and cryptocurrencies can be digitally signed by the HSCA- certificate or trust chain thereof, preventing hacking, cloning, duplication, or fraudulent use.



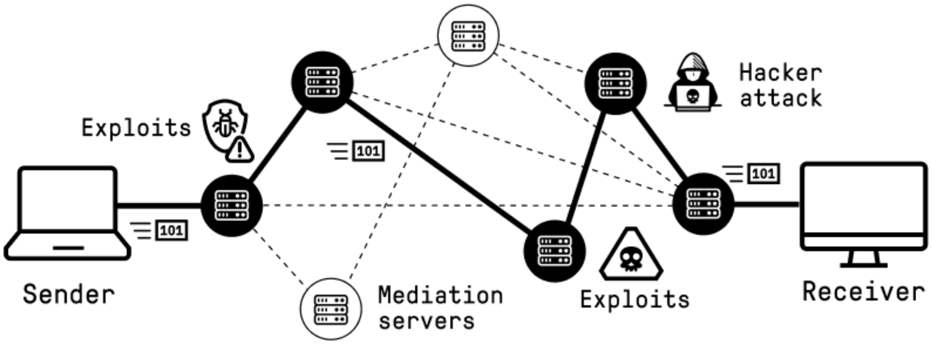
Another key feature of the HyperSphere is its unique form of storing data. Unlike conventional database records stored in large server farms, which are frequently hacked exposing hundreds of millions of people to identity theft, the HyperSphere employs disaggregated data storage of files in diffuse data clouds. In the HyperSphere, only parties participating in the process of creating or updating a database have the ability to gather, aggregate, and restore the original data into legible form. By linking disaggregated file and records storage to HSCA-certificates, the HyperSphere provides tools indispensable in protecting user privacy, preventing unauthorized access to accounts and files. StealthData, now in development, represents a secure data storage service using the HyperSphere's unique features of disaggregated data storage.

Together the SDNP protocol and the HyperSphere platform represent the greatest advancement for fintech since the advent of the Internet and online e-commerce. The migration of financial applications onto the HyperSphere platform is projected to greatly diminish the frequency and magnitude of banking crime and credit card fraud.

Unstoppable Criminality in Today's Financial Transactions

The frequency of financial crimes perpetrated against online and credit card transactions have reached epidemic proportions. Security experts are no longer able to contend with the myriad of attack stratagems employed by an ever increasingly sophisticated and growing pool of cyber criminals. Transactional attacks of theft and fraud can be largely divided into five categories, namely

- Network attacks
- Imposter attacks
- Identity theft
- Edge device attacks
- Database exploits



Each of these methods exploits vulnerabilities intrinsic to present day public and private computer and communication networks and devices we use to perform the commerce of daily life.

In *network attacks*, often referred to as *man-in-the-middle* attacks, a cyber-criminal intercepts communication between transacting parties, either between a buyer and seller, or between either party and a bank executing the financial process, using that information either to corrupt the fidelity of the ongoing transaction or to open a new fraudulent transaction.

In *imposter attacks*, a cyber-criminal pretends to be one of the transacting parities, engaging in the purchase or transfer as if they are operating as a legally authorized agent and a valid party to the transaction. Oftentimes, imposter attacks are performed in the future, not at the time the hacker first gain the trust of the other parties. Once obtaining access, the hacker bides their time, participating in a number of transactions with no hint of hostility, waiting for the right opportunity to strike and maximize their financial return.

In *identity theft*, a cybercriminal executes either a network attack or imposter attack, often performed in concert with *profiling*, not to steal assets, but to obtain information about the account holder or transacting party. In profiling, the cybercriminal studies their victim, their behaviors, activities, social media, in an attempt to discern login passwords and account names. Once they have obtained sufficient information, they may steal their targets assets either all at once (emptying the bank account), or over time in an undetectable manner stealing small amounts of money again and again.

In edge device (malware) attacks, the cybercriminal attempts to invade a software-based physical device such as a phone, computer, or point of sale terminal. To successfully launch a device attack, the perpetrator must convince the edge device to trust them to accept an email, message, file, or application containing malware (easy to do) whereby the malware installs itself into the device much like a biological virus infects a living cell. In the Internet and in computer networking, trust is conferred by a digital credential called a CA certificate, where the acronym CA stands for "certificate authority", a third-party trusted to represent the identity credentials of a transacting party. If a file is sent with a valid CA-certificate, the user will accept the file without suspicion of the embedded malware lurking within its payload.

Some malware infections are insidious, hiding in the device collecting information, creating backdoors, and commandeering control piece by piece until the time is right to strike. More evolved viruses posses the ability to infect other devices in the same network, using each infected to convince other devices to "trust me" when they exchange files or execute shared processes. An especially sophisticated malware attack vector (called Frankenstein viruses) is delivered in pieces of innocuous code appearing to virus checkers as meaningless non-executable junk software. Once the pieces are delivered the malware autonomously assembles itself into a virulent form and launches its attack completely undetected by edge device protective means.

Some malware attack immediately, so called "zero day" exploits executing malfeasant activities from the moment they arrive in the device. Stuxnet, one of the world's most prolific zero-day malware attacks, infected computers across the globe using a valid CA-certificate stolen from a unknowing computer tech company (see the documentary movie Zero Days available from HBO to learn how). Other malware called time bombs wait to attack until certain conditions are met (for example till when a bank account balance exceeds a specified balance). Still others make their presence known by threatening the device owner to pay the attacker to regain control of their device (so called ransomware).

In database exploits, a cybercriminal utilizes the fact that banks are (for reasons of privacy) unable to synchronize their files to exploit inconsistent and asynchronous records. Legally prohibited from sharing client and account information with other banks and financial institutions, cybercriminals have developed strategies to conduct multiple transactions concurrently involving the same assets. Although three day-holds for transaction settlements thwarts the most egregious account abuses, vulnerabilities still remain, especially involving credit card and online transactions.

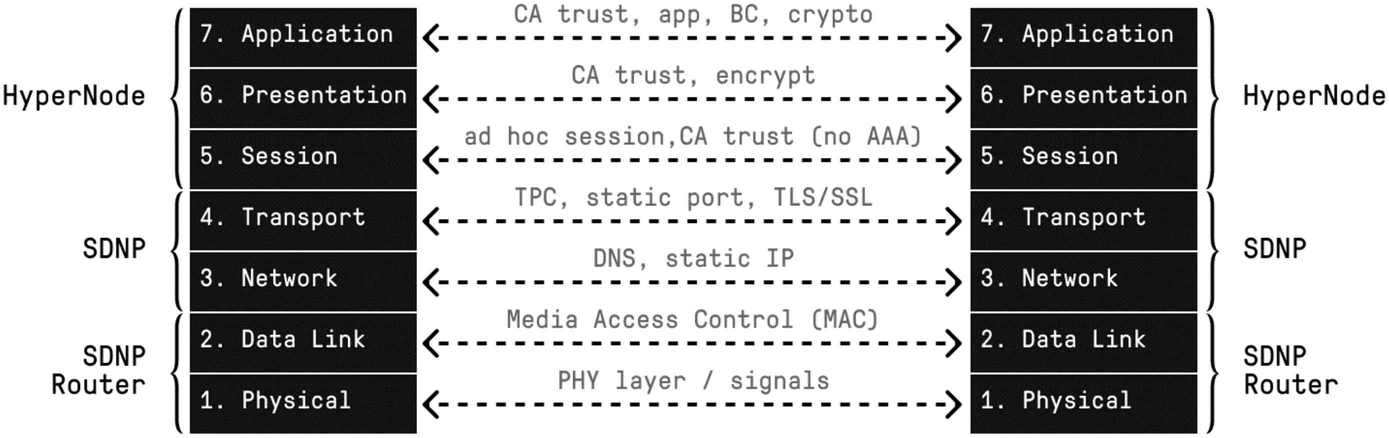
Network Device Interoperability is Security's Achilles Heel

Although methods vary, one commonality of all device and networks today is their reliance on TCP/IP, the transmission control protocol / Internet protocol standard used ubiquitously to move digital data through package switched networks. Since authentic users must exchange information over the network using a publicly defined open-source protocol created to guarantee interoperability, attackers can develop and attempt new attacks in perpetuity without detection, honing their skills until they find a method that works. The Internet and its TCP/IP protocol were never designed to anticipate or repel such invasions.

In order to combat cyberattacks, the Internet and private networks both rely on cryptography to secure communication and transactions. A cipher is a coding that converts digital content (called plaintext) into interpretable data (called ciphertext) using a defined algorithm and one or more pieces of information called a cryptographic key. Since only the transacting parties are privy to the key's content interlopers (at least theoretically) cannot decipher the file's content because they don't have access to the key.

The same method is used to confirm a party's "identity", i.e. using cryptography to create a CA-certificate, whereby the valid ownership of the CA-certificate can be confirmed by the other party. In one such exchange called PKA or public key authorization, the issuer (e.g. a bank) passes a encryption key in public to a buyer, which in turn uses the key to encrypt a test file, and sends it back to the issuer. If the issuer is the true key holder, they will be able to open the file and confirm to the buyer they hold the corresponding decryption key, which remains private. In this and other similarly key or digital certificate exchanges, a transaction should be secure.

In TCP/IP based communication this key exchange occurs at multiple levels. In the Transport layer 4, the OSI layer that defines the method (handshaking) of file delivery in a network, security employs standards such as SSL and TLS. Keys or digital certificates may also be exchanged as part of Session layer 5, where two transacting parties exchange keys to commence communication. Any party not present during this initial exchange will be unable to open the encrypted portion all subsequent data packets exchanged between the parties. Such an exchange may, for example, occur as part of a login sequence to access personal files or bank records. In addition to the foregoing, many companies also execute their own private cryptography as part of the OSI Application layer 7, using algorithms and key exchanges unique to the bank, their clients, and their authorized agents (e.g. point-of-sale vendors).



With these multiple layers of security using cryptography, the Internet, private networks, and financial transactions should, ostensibly, be secure, private and unhackable. Obviously, this is not the case. Financial transactions are routinely corrupted, assets absconded, identities stolen, and devices infected. Clearly cryptography is not achieving the desired goal of delivering a safe, secure, trustworthy, and private platform for financial transactions and commerce.

In fact, innumerable methods have been devised to circumvent cryptographic security without the need to break a code using brute force calculations. These strategies, or exploits, rely on gaining access gradually. Effective cyber-hacking is simply a game of patience, waiting long enough, till a user reveals some information then exploiting the security crack to break in to processes and files in a step-by-step manner. Such as attacks include packet hijacking, man-in-the-middle attacks, denial of service attacks, and fraudulent CA-certificates.

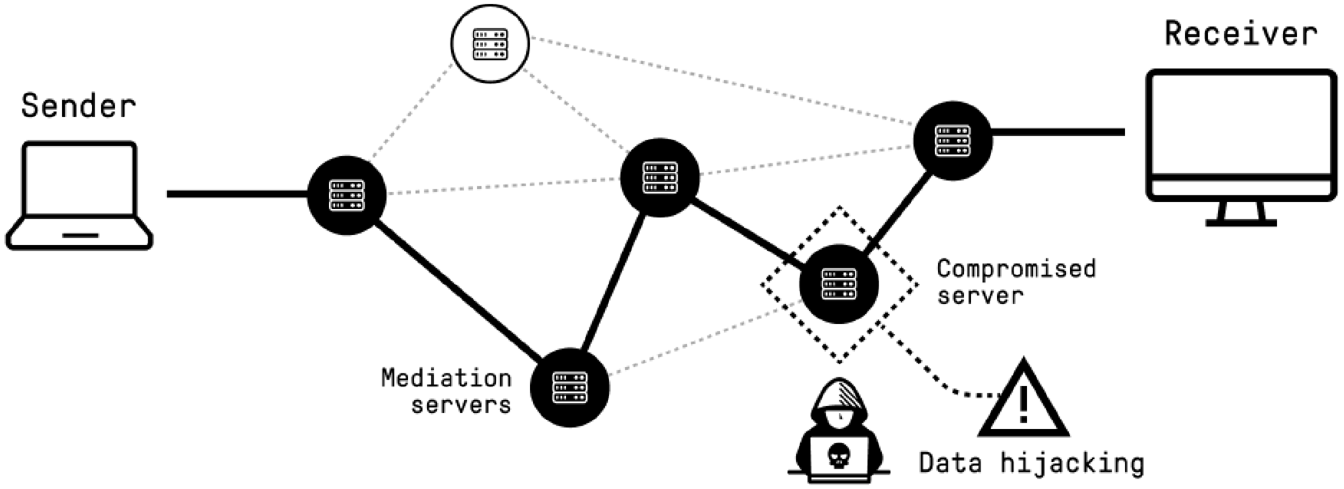
Even worse, SSL/TLS, the core of Internet security, is known to be subject to numerous exploits including the notoriously successful cyberattacks including Unholy PAC, BEAST, Heartbleed, POODLE, DROWN, CRIME, and BREACH, all stole account passwords, server certificates, and other sensitive data. Their success serves perpetually as a painful reminder of what cybersecurity experts quietly concluded long ago – the TLS protocol was never really that secure in the first place.

Why are all these so-called secure processes so easily overcome? For one, TCP/IP was never created to repel cyberattacks because its purpose was simply to ensure data packet delivery over an unstable (or defenseless) communication grid, not to perform financial transactions. The Internet's widespread use in commerce didn't materialize till the advent of the World Wide Web years later.

One particularly problematic security issue is that even when a SSL/TLS vulnerability is exposed and the software patched, many users are either unaware of the new release, or choose not to download it for fear of causing network problems for their user base. So even if one party upgrades to the newer version, if the other party does not, the entire communication or transaction remains at risk for both parties.

Beyond cryptography issues, a fundamental weakness of TCP/IP packet security is the data packet's metadata already tells a hacker all about the packet, who is it from (IP source address), where is it going (IP destination address), what kind of service it involves (port number), and what portion of the data packet is the payload. In cloud-connected devices, the data packet also reveals the local source and destination MAC addresses. Hackers use metadata to identify and surveil packets in same session and involving specific parties.

The other major problem with TCP/IP is a user does not control the routing of the data packet the send. Instead packet is determined by routers and host servers of the clouds and network the packet traverses. And because the IP destination is known, a cyber-criminal can hijack data packets and reroute all the packets of a session through their own server, storing them for later analysis. If these packets contain credit card, personal identity, account information, or login data, financial security is dangerously compromised.



Yet another issue with the Internet is its heavy reliance on CA-certificates and key exchanges for confirming the identity of a transacting party. These identity credentials determine a user's file access and transactional authority, including operating kernel and root access to a host device and hardware. Any corrupted certificate or stolen key exposes both parties in a financial transaction to fraud, theft, and possibly blackmail. Networks communicating with TCP/IP have no mechanisms to confirm user CA-certificate validity or to identify and quarantine stolen security credentials.

Despite, hype to the contrary, blockchains cannot protect the Internet or secure cryptocurrency transactions because the blockchains exist atop of TCP/IP, meaning the network's vulnerabilities are also the blockchain's vulnerabilities. This intrinsic weakness is exemplified by the rampant theft of cryptocurrency coins and wallets over the Internet.

THE HYPERSPHERE DOES NOT RELY ON TCP/IP (OR SSL/TLS)

Data communication in the HyperSphere does not rely on TCP/IP for routing or SSL/TLS for security. Although its data packets follow TCP/IP's format, i.e. they look the same, they do not operate in the same manner.

Instead, a new communication protocol called the *Secure Dynamic Network & Protocol* or SDNP is used to transport data. Its unique network features include several fundamental principles for delivering superior network and device security, comprising:

- Anonymous data packets
- Dispatcher based routing
- Dynamic stateless operation
- Dynamic meshed transport of fragmented packets
- Dynamic security (hop-by-hop packet concealment)
- Fully decentralized network operation

Because of these unique multi-factor security provisions, Layer-4 transport security protocols (such as SSL, TLS, or IPsec) play a diminutive role in the HyperSphere's superior security and data integrity. As such, the hypersecurity conferred by Secure Dynamic Network & Protocol does not depend on cryptography or a single defense mechanism to repel cyber-attacks and confound hackers.

Anonymous Data Packets

> HyperSphere datagrams are anonymous containing the dynamic IP address only of the next single hop destination and revealing no information regarding a packet's sender or receiver

In the HyperSphere, several provisions are employed to obfuscate the identities of transacting parties by employing *anonymous data packets*. Although the packets follow the same 7-layer OSI model as a conventional TCP/IP packet, the HyperSphere data packets do not reveal the IP address or true identity of the transacting parties. In particular, SDNP packets never contain the IP address of the ultimate destination of the packet. Instead a HyperSphere's datagram shows the destination IP address of only one hop, the IP address of the next destination, and no farther. Without knowing where the packet is headed, metadata monitoring is meaningless and user profiling impossible.

Dispatcher Based Routing

> HyperSphere packet routing is performed by an independent dispatcher function (authority nodes) with no access to the content of the data being routed. Datagram transport is performed by task nodes with no idea as to the ultimate destination of packets or payloads

But without a destination IP address, how can a HyperSphere datagram reach its ultimate destination, the edge device of the other transacting party? To solve this problem, the HyperSphere borrows a concept from the world of professional communication – the dispatcher. Specifically in the HyperSphere's dispatcher based routing, SDNP software called "authority nodes" schedule the routing of all network traffic through the HyperSphere. Using a set of computers always separate and distinct from devices hosting HyperSphere task nodes carrying content, authority nodes issue only command and control (C&C) packets instructing task nodes as to packet routing. Authority nodes never carry transactional or media content, or even have access to it.

During transport, task nodes participating in a transaction or call are warned in advance by the authority node to be lookout for an incoming packet having a specific identifying tag. The C&C packet instructs each task node what to do with the tagged packet when it arrives, specifically how to process it and where to send it. So each node in the HyperSphere only knows information regarding a single hop, i.e. where to send the datagram next.

Metaphorically, HyperSphere network operation is much like a freight shipper where the office in Hong Kong is instructed to send a specific incoming package to Tokyo, and where the office in Tokyo is told to forward the expected package, once received, onto to San Francisco. The Hong Kong office, however, has no idea the package is ultimately destined for San Francisco or beyond.

Dynamic Stateless Operation

> Nodes performing tasks or issuing commands in the HyperSphere are stateless having no record of prior tasks they performed or of packets delivered (i.e. with perpetual amnesia)

Aside from employing dispatcher-based routing, the HyperSphere's authority and task nodes are dynamic and stateless, meaning they change constantly, retaining no knowledge or history of their prior actions. Dynamic stateless operation employs time as a weapon to limit the opportunity window and surface of a cyberattack, in so doing confounding hackers and frustrating cybercriminals. Every fraction of a second, cybercriminals must restart their attack anew.

In dynamic stateless operation, once an authority node schedules a datagram's routing, it immediately forgets what is decided or commanded. Likewise once a task node receives, processes, and forwards a datagram containing transaction or media content to its next target, the node immediately forgets all information about its actions, retaining no copy of the packet, its payload, or even that it processed it. In essence the HyperSphere is a dynamic communication network made up of software nodes suffering perpetually recurring amnesia and selective dementia.

As a further degree of dynamic security and anonymity, all IP addresses used in HyperSphere datagrams are supplied by the HyperSphere's own name server, not by the Internet's DNS servers. The exclusive use of HyperSphere name servers prevents name-based misdirection and packet hijacking.

The HyperSphere name servers operate in two different ways depending on whether communication is occurring over a private network or in a public cloud. Within a private network, the HyperSphere name server function much like a dynamic host configuration protocol (DHCP) server, dynamically assigning temporary IP addresses to devices or servers whenever they join the network and using these addresses until they change. Even if the same address is used on another network or on the Internet, the identified devices in the private cloud will not be the same as on the public network, and essentially unreachable (except through a portal or NAT). In banking, for example, a private cloud may be used to access ATM machines, with no connectivity to or access from public clouds.

In public TCP/IP clouds or over the Internet, the HyperSphere behaves much like an overlay network, coexisting with TCP/IP but managing routing by giving each TCP/IP router instructions likely requiring only direct single-hop datagram transport. In such cases the HyperSphere name server acts like a dynamic network address translator (NAT), converting SDNP specific addresses to IP addresses assigned by other name servers ad network operators. Whenever a device joins or leaves the HyperSphere, the HyperSphere name server is updated. In this manner most IP addresses are likely dynamic with no identity associated with them.

Dynamic Meshed Transport of Fragmented Packets

HyperSphere network traffic is carried over perpetually changing paths of a time-variant dynamic meshed (DyDAG) network absent any predictable routing. Payloads are parsed into fragmented data snippets lacking meaningful content and transported over different routes.

Another unique feature of the HyperSphere is its pioneering application of routing communication sessions by operating the network as a 3D dynamic directed acyclic graph or DyDAG. In mathematical graph theory, a directed acyclic graph or DAG comprises a graph in multiple dimensions where no two successive packets traverse the same path or form a closed loop.

Adapting the DAG concept to routing data in the HyperSphere, packets dynamically change routes constantly traveling across the network using an ever-changing group of task nodes. Managed by the HyperSphere's authority nodes, datagram route selection is dynamic, made to (i) minimize propagation delays for the best real time performance and the highest quality of service (QoS), and (ii) chosen to avoid sending packets from a specific session or transaction repeatedly through the same task nodes. Surveillance of specific computer or network produces no useful information because of data passing through any one node and host device is extremely sparse and limited.

Routing of datagrams that change over time is referred herein to a *dynamic meshed transport*. As such datagram routing over the meshed graph is dynamic, i.e. changing with time. The HyperSphere's inventors refer to this new kind of state-based meshed network traffic management as a DyDAG, an acronym for a dynamic (state-based) directed acyclic graph.

The beauty of 3D DyDAG routing is that path optimization includes local propagation delay information not known by any central authority. Employing a distributed or tiered hierarchy of authority nodes, no node in the network is ever aware of the fully routing of a packet. Accordingly, no one (including the network owner or operator) can subvert the autonomous dynamic routing of the network. And since no person or computer has *a priori* knowledge of a packet's route, there is no benefit in blackmailing or bribing a network operator to gain control or insight into data transport. If no one has meaningful knowledge as to packet routing there is no reason to subvert anyone.

Not only does the HyperSphere employ meshed DyDAG routing of its datagrams but also no single packet carries meaningful content in its payload. Instead before routing commences, the content of a message, transaction or file is fragmented, i.e. parsed into small pieces of data snippets. Each of these data snippets are loaded into the payload segment of multiple SDNP datagrams and sent across the meshed network over different paths. Fragmented data transport means successfully intercepting a packet in transit produces no useful information.

Successfully intercepting a sufficient number of packets to reconstruct the original content is virtually impossible because (i) no one knows the path the packets containing fragmented packets will take, (ii) the anonymous data packets make it impossible to identify which packets related to the same session, and (iii), even if you could find which packets to intercept, because of the network's dynamic stateless operation, by the time a computer can be identified and attacked, the datagram of interest is gone and all history erased.

Dynamic Security (Hop-by-Hop Packet Concealment)

HyperSphere packets employs dynamic security of payloads, where transactional, data, and media content are dynamically concealed by perpetually changing algorithms and state-based security credentials including scrambling, encryption, junk data, mixing, and splitting.

As if dynamic meshed routing of fragmented data in anonymous datagrams carried by stateless nodes were not tough enough, the SDNP also employs *dynamic security* provisions, algorithms and state-based security credentials concealing packet content that change over time and on a hop-by-hop basis.

In operation, each task node processes incoming and outgoing packets in accordance with state-based security credentials and payload concealment methods, including dynamic unscrambling & scrambling, dynamic decryption, and encryption, dynamic junk data insertions and deletions, and other methods. Security credentials and packet concealment change hop by hop with no central authority and no master decryption keys (decentralized autonomous operation). Collectively, this process is referred to as packet concealment because without the proper numeric seeds and cryptographic keys it is impossible to figure out how the packet's payload was modified.

So even in the unlikely event that packets from the same session are routed through the same node, the security credentials (keys, time, tags) used to encode and conceal the packet's payload are state based, and will have changed. Moreover, some dynamic security protective provisions change on a hop-by-hop basis, where only two communicating servers share information needed to recover a packet's content. No master key or network operator is involved. Moreover, by combing scrambling with cryptography, it becomes impossible for even quantum computer to know when it has successfully broken a cryptographic code, because the resulting plaintext file is scrambled, still appearing as ciphertext or useless gibberish.

Fully-Decentralized Network Operation

The HyperSphere operates as an autonomous fully-decentralized network having no owner, network operator, or master key able to subvert its secure operation. Packet routing occurs autonomously using current network performance criteria unknown by users or resources.

As a final protective provision in securing HyperSphere communication, the network itself operates as a fully autonomous and decentralized cloud with no owner, operator, or master key holder to beneficially subvert, rob, or coerce.

Any device joining the HyperSphere's cloud has the opportunity (with appropriate compensation) to carry small amounts of data snippets, perform limited distributed computing calculations, or store fragments of disaggregated data as part of a diffuse cloud. Devices or software lacking the proper security credentials are barred from joining the network, whereby any packets they send will be ignored and discarded.

In essence, the HyperSphere comprises a global dynamic *ad hoc* heterogeneous network of software nodes installed on servers, PCs, vehicles, IoT, and mobile devices able to communicate over any available physical medium capable of packet switched datagrams including Ethernet, Wi-Fi, mobile cellular networks (2G, 3G, 4G, 5G), DOCSIS3 or other fiber protocols. In the absence of a fixed radio cellular or WiFi network, nodes will automatically establish a peer-to-peer network with other devices if possible, operating as a fully autonomous *ad hoc* network until client-to-radio service is restored.

One of the most powerful security features of the HyperSphere's fully-decentralized cloud is that the dynamic concealment of payloads changes every fraction of a second, occurring on a peer-to-peer single-hop basis, i.e. *sans* network operator participation.

As illustrated in the hit movie Imitation Game, Professor Turing, the inventor of the modern-day computer, had only twenty-four hours to break the Enigma code each day before the cryptographic key was reset and he had to start all over again. In contrast, HyperSphere hackers have only a fraction of a second to break more than 100-years' worth of encryption.

Even in the amazingly unlikely event a mythical hacker employs quantum computing to crack a specific packet's encryption and break its dynamic concealment, the cryptographer will be extraordinarily frustrated to find the content of the packet is on its own, meaningless and totally useless. Only by finding all the other related packets from the same session traveling through the HyperSphere at that time, concurrently cracking all of them, and reassembling the fragmented content together can any useful content be extracted. But since the packets travel anonymously over a myriad of ever-changing meshed network routes, there is no way to locate the missing packets.

The ultimate protective provision of the HyperSphere securing its packets is the speed of light and time itself. Since in fragmented data transport, datagrams propagate an ever expanding volume as the move through the cloud, even if a hacker magically knows where related packets will be traveling, by the time their attack vectors reach a targeted server, the packet they seek to intercept will be gone and all history of the packet lost forever. A hacker cannot catch a packet by traveling at the same speed as the packet it's chasing.

Collectively, the foregoing communication methods of the HyperSphere and its SDNP protocol deliver *hypersecurity* to clients seeking to protect financial transactions from attack or surveillance., protections the Internet, TCP/IP and SSL/TLS can never achieve.

PROTECTING PRIVACY & CONTROLLING DATABASE ACCESSIBILITY

Although the HyperSphere delivers superior protection of transactional processing for communication and network operations, protecting a user's privacy and personal information requires additional mechanisms to prevent imposter attacks and unauthorized access to database records. In this regard, the HyperSphere offers two unique features, namely:

- Network native HyperSphere certificate authority (HSCA)
- Diffuse cloud storage of disaggregated data

Network Native HyperSphere Certificate Authority (HSCA)

> To ensure personal privacy, the HyperSphere operates as its own certificate authority generating HSCA-certificates combining hashed data proving personal identity and cryptographic signatures unique to the HyperSphere network thereby preventing the issuance or use of fraudulent certificates to subvert database login access or fraud.

Unlike the Internet, which relies on digital CA-certificates and trust chains issued by third parties subject to fraud and theft, the HyperSphere includes its own internal mechanism to generate network-native digital certificates, i.e. HSCA-certificates, to validate personal identity and to control database and device access privileges.

Because the HyperSphere participates both in issuing the HSCA-certificate and in validating a certificate's authenticity using AAA (authentication, authorization, and administration) mechanisms, there is no chance that an imposter can subvert the authentication procedure. So long that the personal identification documents used to open an account are valid, the resulting HSCA-certificate generated is incorruptible because only the HyperSphere network knows the network identity used to digitally sign the issued HSCA-certificate.

The veracity of the personal identification documents used to open an account or engage in a transaction depends on the KYC/AML verification process of the financial or governmental institution performing the background check. Provided the identity check is authentic, the resulting HSCA-certificate cannot be faked or imitated by an imposter without flagging a personal identity mismatch, a network credential mismatch, or both.

Once the personal HSCA-certificate is generated, a verifiable ancestral trust chain from the parent identity certificate can be used to generate valid root and leaf (issuing) HSCA-certificates. These certificates can in turn be used to sign devices, register software installations (to prevent cloning), create logins, open bank accounts, establish database access privileges, control read and write access to private blockchains, and create privately owned crypto-wallets. Attempts to steal accounts, clone devices, or perpetrate imposter exploits protected by HSCA-certificates will fail to pass network AAA validation, and criminality thwarted.

As an added benefit descendant HSCA-certificates may employ pseudonym identities, allowing legitimate and binding legal transactions to be executed without exposing the true identities of the transacting parties, but where a financial institution and government regulators are still able to confirm the parties' true identities. Such benefits are especially important to protect the privacy of buyers and sellers in multimillion-dollar transactions involving high-value assets such as gems, jewelry, oil, art, precious metals, and cryptocurrency where extortion, blackmail, kidnapping, and violent acts of malfeasance are potentiated.

Diffuse Cloud Storage of Disaggregated Data

> To ensure protect against unauthorized access, theft, or content modification of personal or corporate databases, the HyperSphere employs disaggregated data storage over diffuse data clouds where no one device holds meaningful content, thereby thwarting all benefits of hardware-based attacks. Database access, including both relational databases and private blockchains, is limited to valid HSCA-certificate validated account holders and their agents.

Once of the greatest risks to personal privacy today is the hacking of private and commercial databases. Corporations, government agencies, banks, credit bureaus, and other financial institutions following the premise that storing all their information in one place and putting a firewall around it "makes it more secure" have been hacked time and time again, resulting in the personal information theft of hundreds of millions of clients and citizens.

This naive IT practice is predicated on the principle that file encryption and login protection can protect against unauthorized access to a database. Both assumptions are incorrect. Another problem with concentrating data is the entire database must be backed up in a second location for emergency recovery purposes, doubling the exposed attack surface of the data. Moreover, oftentimes, backup facilities are much less secure and even more concentrated than files in their original source computer network.

In the HyperSphere, database security employs precisely the opposite method, one where data is spread over as many devices and locations as possible, making it impossible to locate the location of a file because it is not localized. Only those users with authorization receive the links and keys needed to recall a relevant portion of the database. No master key is needed in a disaggregated database so no ambitious cyber attacker is able to access the entire file structure even if they some how gain unauthorized access to a portion of the data structure.

Coupled with access control controlled by multifactor HSCA-certificates and AAA validation further ensure security and privacy. And because the fragmented data is stored in redundant copies spread across a diffuse cloud, no specific backup file concentrating the records is required.

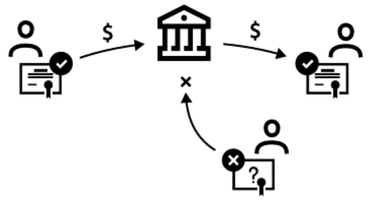
The same protective mechanisms control access to private blockchains, wallets, and bank account records.

Benefits of the HyperSphere in Finance and Fintech

The HyperSphere's unique approach to securing communication and protecting privacy confers numerous unique advantages in financial transactions, including:

- Securing & privatizing bank wires
- Preventing invalid online transactions
- Enabling secure bank-client messaging
- Preventing personal Information database theft
- Preventing fraudulent credit card purchases
- Protecting against unauthorized account access

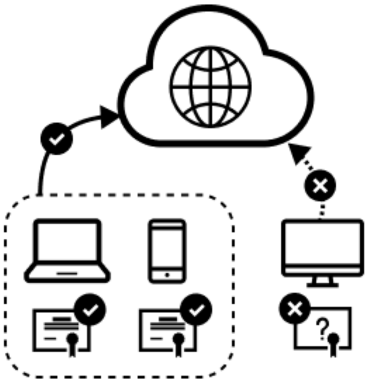
Securing & Privatizing Bank Wires



Using a bank HSCA-certificate and a key exchange involving the HSCA-certificates of a money wire's sender and recipient, only the true intended recipient of a money wire has the ability to collect the transferred funds.

In the event of an incomplete transaction, the processing bank returns the funds (less fees) to the sender with no risk of loss of the asset (unlike today where sometimes the wired money is never returned).

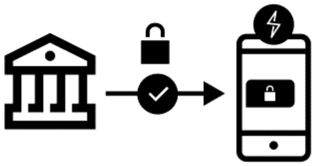
Preventing Invalid Online Transactions



In the HyperSphere all personal devices including smartphones, tablets, notebooks, desktop computers, gaming consoles, and IoT enabled appliances and assistants are registered onto the HyperSphere network using the owners personal HSCA-certificate. Fraudulent online purchases executed through the HyperSphere can be immediately rejected when the device used to place a purchase is not a validated HyperSphere device or node.

In the event that a purchase is made on another device, the HyperSphere client is able to validate the transaction via a login-based application software and device signed by an HSCA-certificate

Enabling Secure Bank-to-Client Messaging



Personal messengers carrying text and VoIP over the Internet are untrustworthy and unsecure. These messengers, even Telegram, are notorious for their ease of being hacked, monitored, or surveilled.

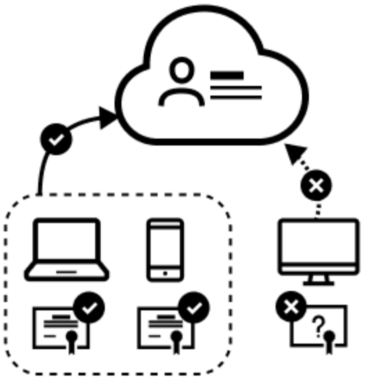
The SDNP based messenger, StealthTalk, meets FIPS140-2 military grade security standards, providing both secure communication, and identity-based access and privacy provisions. StealthTalk may be used on a stand-alone basis or may be integrated into bank-customized applications.

Preventing Personal Information Database Theft

The HyperSphere's disaggregated data storage over diffuse clouds renders data breaches and renders computer server farm attacks useless as no meaningful data is concentrated in any one device or geography.

AAA validation of HSCA-certificate holders to confirm identify beneficially provide a network intrinsic mechanism to define and limit read and write access to files and databases containing personal information of clients.

Protecting Against Unauthorized Account Access



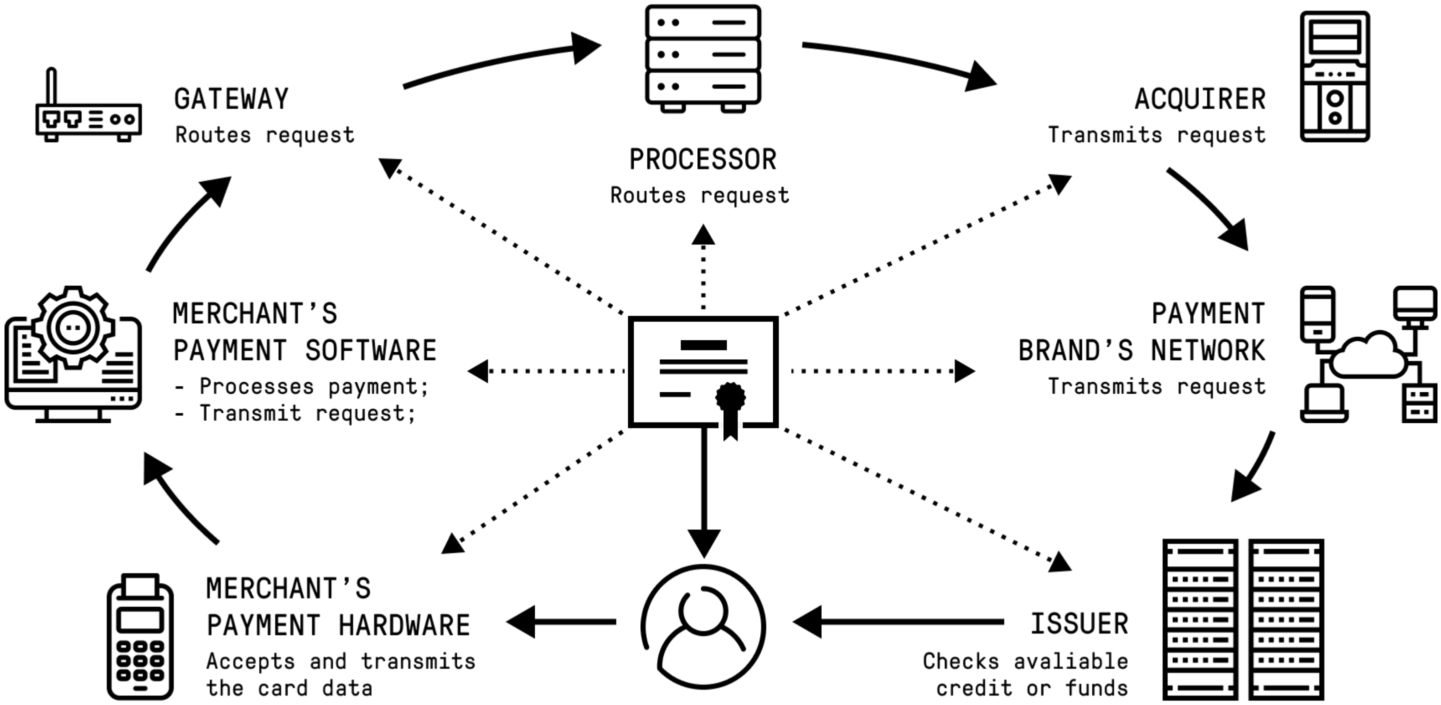
In the HyperSphere all personal devices including smartphones, tablets, notebooks, desktop computers, gaming consoles, and IoT enabled appliances and assistants are registered onto the HyperSphere network using the owners personal HSCA-certificate.

Login access to files, database and personal information may be restricted to only sessions launched over a HyperSphere registered device and using application software and device signed by an HSCA-certificate.

Preventing Fraudulent Credit Card Purchases

Because of the large number of parties involved in processing a credit card payment and point-of-sale (POS) purchase, the possibility of fraudulent purchases is great. Conversely, by imposing strict security measures, the possibility of a valid transaction being rejected and causing a cardholder undo burden and personal cost in confirming proof of identity is equally problematic.

The HyperSphere prevents these complications in several ways. Firstly, all agents and devices are digitally signed by HSCA-certificate meaning only valid registered participants can engage in a transaction. Secondly, the purchaser's card can be linked to own HSCA-certificate. During the transaction, the process could be automatically verified by confirming the user's registered mobile device is in the same location as the POS transaction, and if not by using an link to the user allowing them to validate the transaction via a login confirmed application.



Transactions executed using data transport over the HyperSphere are much more secure than those performed over an analog phone line (modem) or the Internet.