Whitepaper

Liechtenstein Protocol

Standardized Framework for the Tokenization of Securities
Version 1.0 - March 11, 2021

Monty C. M. Metzger
Anurag Verma
LCX.com

Abstract

Liechtenstein has introduced a new legal framework called the Token and Trusted Technology Service Provider Act (Officially: Token- und VT-Dienstleister-Gesetz; TVTG) also known as the Blockchain Act. The regulatory framework enables a technologically neutral token definition building the foundation for a tokenized economy. The new laws bridge the gap between the digital and the physical world and define key roles along the value chain of tokenization. While the new legislation has been implemented, there is a need for a new security token standard to implement compliance and regulatory requirements directly on technology and on the token level.

Our technical whitepaper is a proposal of this new token standard as an industry-led initiative, building upon the basic experience and knowledge of other standards such as ERC1400, ERC1404, R-Token, or T-REX. The new token standard is blockchain agnostic and will be implemented on a variety of public chains. The blockchain agnostic standard aims to build a framework for the tokenization of securities and to foster the growth and proliferation of asset tokenization as a practice.

The Liechtenstein Protocol standard, and its first iteration, is programmed to automatically enforce specific conditions that relate to legal and regulatory requirements applicable to securities and enables automated compliance of the tokenized asset with pre-defined requirements built in the code.

TABLE OF CONTENT

1. Introduction

Historically, transactions have depended on processes that build much-needed trust—that is, each party must offer evidence that it has the capacity to hold up its end of the bargain. If one party hands over the cash and the other party fail to deliver the security, serious market inefficiencies arise. Financial intermediaries— whether for commercial reasons or due to regulatory mandates—have blossomed in every place that market integrity is lacking, in order to provide assurances that the markets will function without undue friction. The blockchain can resolve that issue.

For the first time in history, an immutable, decentralized ledger exists on a global scale, eliminating the need for middlemen, complex auditing systems, and long settlement times. Open (permissionless) protocols mean settlements no longer depend on connecting fragmented legacy systems. Additionally, because the ledger is append-only (existing records are immutable), it provides a high degree of accountability, with blockchain time-stamping built-in. In other words, a reliable audit trail is built into the technology.

Over the past few years, turing-complete programming languages have been implemented into decentralized blockchains. These systems use “smart contracts” (software programs stored on-chain that are automatically implemented upon specific conditions being satisfied), to add and modify data algorithmically however a user designs it. This data extends well beyond simple account balances and may include metadata, account restrictions, transfer rules, as well as any other calculations a regular computer can perform.

The most widely used turing-complete blockchain, Ethereum, grew out of frustration with trying to implement complex logic on top of Bitcoin. Ethereum simplifies the task of implementing complex financial logic on a blockchain. With only a few lines of code, smart contracts can transfer assets or establish escrow conditions to be executed algorithmically, with all the benefits of blockchains as described earlier. In other words, if two parties enter into a smart contract, and each party presents their asset, the transaction is automatically effected without risk of failure; if one party fails to present its asset, the other party retains its asset and can move on. There is no risk of payment on one side, and the failure to deliver on the other side. The smart contract can be designed to affect a transaction instantaneously or can be designed to affect upon future conditions begin met.

Recently, smart contracts have been used for Initial Coin Offerings (ICOs) or token sales. These tokens usually conform to a standard (e.g. ERC20 or ERC223, which are Ethereum-based technologies that generally facilitate the exchange of tokens of the same standard), which allows them to be offered for sale and trade on a number of online platforms.

The global adoption of various forms of token sales and ICO fundraising structures has led to an explosion of new capital formation—whether through virtual currencies (representing a means to transfer value but not backed by other functionality or a promise), utility tokens (software that might give access to goods or services, rather than being designed as a financial instrument), or tokenized securities like equity in companies—that has outpaced both the seed and venture capital investment markets. ICOs and other sales have raised USD $2.3 billion to date. In 2017 alone, ICO funding surpassed USD $1.2 billion. Unfortunately, some ICOs have made inaccurate, and in some cases, fraudulent claims in an attempt to raise funds. This has attracted the attention of regulators in a number of countries and pressed them to consider whether to formally bring the investor protections of securities laws or bear on the diverse universe of transferable tokens of various kinds.

There are two basic kinds of tokens sold in a token sale or ICO: utility tokens and security tokens. Utility tokens are used to access services or assets, which are often based on smart contract technology. The purchase of a utility token is akin to purchasing the rights to use software or a product. These tokens are like pay-per-use SaaS offerings, subscriptions to content, or a means to compensate contributors to a platform in a manner similar to in-game currencies. One kind of security token represents an equity stake in an organization, or a claim to the wealth generated by its activities. Sales or issuances of these tokens with these features tend to constitute a securities offering, which means that they are subject to securities regulations. Issuers need to ensure that token sales comply with all applicable securities laws or risk severe penalties. Registrations and exemptions must be considered, and the efficient transferability of tokens that are at the core of their technology may be stifled by a regulatory apparatus that requires intermediaries and government filings of various kinds.

2. Security Token Definition

Traditionally, securities can mean a position of ownership in a publicly-traded company, a creditor relationship with a governmental corporation, or rights to ownership of an asset. A security token is essentially a tokenized, digital version of these traditional securities.

According to European Securities and Markets Authority (ESMA) security tokens fall into a category of crypto-assets. ESMA defines that “crypto-assets are a type of private asset that depends primarily on cryptography and digital ledger technology”.

U.S uses a Howey test to determine whether a token or asset is a security or not. This test came into existence following the monumental case handled by the Supreme Court in 1949. The case was SEC vs Howey which was regarding the need to establish a test to determine if a particular arrangement includes an investment contract or not.

Although originally the Howey Test used the word “money”, future cases included other investments and assets in the test other than money. Additionally, other crucial elements are included to determine a security. Is the profit generated by the investment in control of the investor or not? If not then the asset is commonly declared a security.

These guidelines are relevant in the case of ICOs, STOs, and tokens in the sense that if a token meets all the three mentioned criteria then it is considered a security. These tokens commonly get their value from an external, tradable asset. Also, since these tokens are security, they are subjected to federal securities and regulations. An STO which doesn't follow the regulations will be subjected to penalties.

If the regulations are followed properly, these tokens can prove to have powerful use-cases.

3. Transforming the global securities market

Blockchain technology created an immutable ledger with the potential to transform the private securities market. The technology can simplify the transfer of ownership, create transparency, reduce administrative burden, and provide opportunities for greater liquidity.

With liquidity, investors have the flexibility to sell private securities for an efficient market price, increasing marketability and unlocking greater potential asset value for issuers. There are many asset categories that can benefit, including real estate assets such as limited partner interests in real estate investment funds, fractional ownership in land/buildings, and derivatives.

Liquidity, however, also brings significant regulatory challenges to issuers and investors. Private securities must fall under exemptions with applicable laws to avoid onerous public filing requirements. These exemptions can require limiting the number of total investors, only allowing specific types of investors (e.g. accredited investors), implementing a holding period, and applying many other rules. Restrictions differ by jurisdiction, and compliance with both the issuer's jurisdiction as well as each investor's jurisdiction is mandated. Furthermore, these restrictions apply not only to the initial offering (where much of the responsibility lies on the issuer), but to all secondary trades where responsibility is also placed on the seller. Depending on their specification, tokens may constitute financial instruments subject to financial market law. This may include tokens that have characteristics of equity securities or other investments.

In the U.S., private securities must fall under exemptions within the Securities Act of 1933 as well as the Securities Exchange Act of 1934—and in some cases, the Investment Adviser Act of 1940 and the Investment Companies Act of 1940—to avoid public filing and other costly and operationally prohibitive requirements. For example, in many cases, an owner of limited partner interests in a real estate investment fund can rely on Section 144 of the Securities Act of 1933 to conduct secondary trades. This would require the seller to comply with a one-year holding period or trade with only a Qualified Institutional Buyer under Rule 144a. In addition, the issuer of the initial offering must also continue to be exempt under the Securities Exchange Act of 1933, the Securities Exchange Act of 1934 and the Investment Companies Act of 1940, which may, among other things, require limiting the number of accredited and non-accredited investors.

With the introduction of the Blockchain Act, a new set of blockchain related laws, have been introduced by the Liechtenstein Government. This includes clear regulation on Tokenizing “old-world” assets like equity, stocks, limited partner shares, real estate, derivatives or other forms of securities.

Enforcing these new regulatory requirements has been a significant deterrent in the adoption of blockchain technology for private securities issuances.

The global securities market is composed of three major instrument types: equities, debt, and derivatives. In 2016, these three markets had total notional values of US $67 trillion, $99 trillion, and $1.2 quadrillion, respectively.

In this paper, LCX is laying out a solution to this compliance challenge using the Liechtenstein Protocol. The Liechtenstein Protocol is based on the Liechtenstein Blockchain Act enabling the trade of private securities on blockchains in compliance with regulatory requirements.

4. Tokenizing private securities

One of the primary reasons for issuing private securities is the relative ease and cost-effectiveness of the initial issuance. However, secondary trading of private securities often requires various middlemen (such as brokers and exchanges). In addition, the process for tracking trade activity is manual and costly, and there is a significant burden on issuers to safeguard against potential regulatory risk. These inefficiencies can often lead to issuers imposing trade restrictions, making private securities illiquid. To account for the lack of liquidity, the value of private securities is discounted (i.e. the “illiquidity discount”), preventing issuers from capturing the full value of the underlying asset.

In comparison, public securities can have deep markets and high liquidity, as non-controlling holders can generally resell them freely. However, it is time and cost-intensive to IPO and remains a public company. The process to complete an IPO is complex and can often require 12 to 18 months of preparation. Significant costs apply not only to the offering itself but also to the ongoing process of being a public company, including rigorous regulatory and reporting requirements. In other words, the cost-effectiveness of public securities is low compared to private securities.

By tokenizing private securities, we can potentially increase liquidity and cost-effectiveness. Tokenized private securities (i.e. security-tokens) can be more easily traded on the secondary markets without the administrative burdens of traditional private securities. In other words, tokenized private securities can potentially have more liquidity while maintaining their cost-effectiveness.

Given that the asset categories within the private securities market are massive (in the trillions), and that the illiquidity discount can be as high as 20-30%, the tokenization of private securities has the potential to unlock billions of dollars in value.

In addition to providing liquidity, tokenizing private securities creates opportunities for greater efficiency. Blockchains enable trades to occur securely between two parties without a middleman. It is an immutable ledger where every transaction is automatically recorded and easy to audit. The process and timeline for settlement and clearing of transactions can also be condensed significantly, and any reconciliation processes can be greatly simplified. Tokenizing private securities has the potential to significantly reduce costs, increase the speed of settlement, and improve security. It is compliant with regulatory requirements that now must be addressed in order to enable the adoption of securities on blockchains and transform the private securities market.

5. The Liechtenstein Protocol

The Liechtenstein Protocol addresses the need for compliance on secondary transfers. The Liechtenstein Protocol is programmed to automatically enforce certain constraints that are made a requirement by the legal and regulatory compliances which are applicable to securities. This in turn automates the compliance of the tokenized asset by making it mandatory to meet predefined requirements that are built in the code.

The new token standard as an industry-led initiative, building upon the basic experience and knowledge of other standards such as ERC1400, ERC1404, R-Token, or T-REX. The new token standard is blockchain agnostic and will be implemented on a variety of public chains. The blockchain agnostic standard aims to build a framework for the tokenization of securities and to foster the growth and proliferation of asset tokenization as a practice.

The first implementation is built on Ethereum Blockchain ERC-20, a standard widely supported by the existing blockchain ecosystem. In addition to that, the Liechtenstein Protocol standard will be applied to blockchain networks such as Polkadot, Cardano, ICON, and others.

The Liechtenstein Protocol embeds compliance at the token level and allows for decentralized trading of private securities across any platform that supports the specific blockchain network.

This significantly increases liquidity in comparison to confining trade within a single centralized exchange (the walled-garden approach) in order to enforce regulatory compliance. Centralized exchanges can enforce regulatory requirements to a limited degree, including KYC, AML, and accreditation. However, unless they take a walled-garden approach, once a token leaves that exchange, the issuer may be unable to enforce core securities regulations such as restricting the number of investors, requiring a minimum number of investors, and stipulating ownership levels.

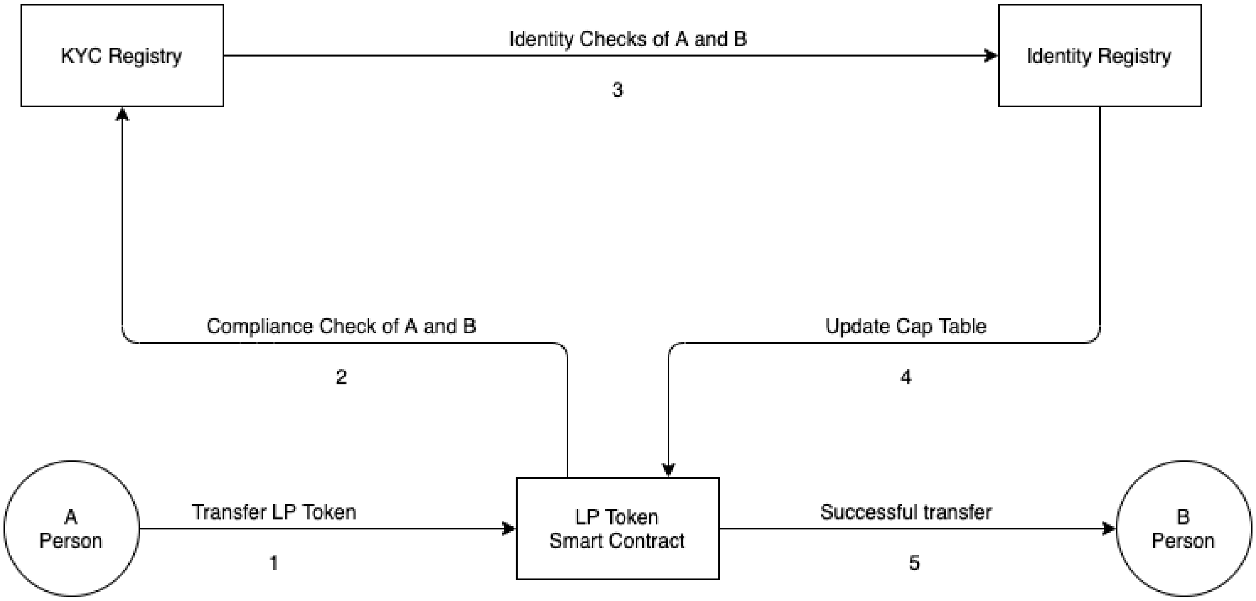

Figure -1 Flow of transfer

The LP security token transfer takes place after a complete compliance check and identity check of sender as well as receiver. In the above fig, Person A will use LP Token Smart Contract to transfer to Person B. If in any of the checks, one of them fails then it will give an error message to Person A.

Our proposed solution enforces regulatory compliance at the token level, thereby meeting core securities requirements regardless of whether the trade occurs on centralized or decentralized exchanges.

The Liechtenstein Protocol includes three core services on the blockchain: (1) Liechtenstein Protocol Token (LP Token), (2) the Compliance Service, and (3) the Identity Registry. While all three services can be implemented as a single smart contract wherein the Compliance Service and Service Registry functionalities are built into the Liechtenstein Protocol Token smart contract itself, this white paper will exemplify an implementation of these services as three distinct smart contracts. Implementation of distinct smart contracts allows for easier upgrades as regulatory requirements evolve over time.

6. Liechtenstein Protocol Token (LP Token)

The Liechtenstein Token (LP Token) is an authorized smart contract that can represent ownership of securities. LP stands for Liechtenstein Protocol.

The Liechtenstein Protocol is blockchain agnostic. For Ethereum the token will be named ELP, the E stands for Ethereum. For Polka Dot the token will be named PLP, the P stands for Polkadot. For Cardano the token will be named CLP, the C stands for Cardano. For ICON blockchain the token will be named ILP, the I stands for ICON.

Token Standards

The first iteration of the Liechtenstein Protocol Token is based on the ERC-20 token standard. It introduces a standard for Fungible Tokens. This implies that they have a property that makes each Token be exactly the same (in type and value) as another Token.

The LP token will be implemented as smart contracts on the various blockchain, ie, ELP to Ethereum Blockchain. It consists of all the standard functions of the ERC-20 token and contains few more features which are relevant to the protocol. Here are the features,

Time Locking

A Time Locking is also known as a lock-in, lock-out, or locked-up period, is a predetermined amount of time following a Security Token Offering. Here, large shareholders, such as company executives and investors representing considerable ownership, are restricted from selling their tokens just like an Initial Public Offering.

A Time Locking is designed to stop early investors and insiders from selling their shares for a set period once a company completes a security token offering (STO), helping to minimize selling pressure in the early stages of life as a publicly-traded business.

Burnable

A Burnable token means the company can repurchase tokens and burn those tokens. They don't possess any financial value and are void of ownership in the company.

Companies will issue security tokens to raise money and expand business operations. Subsequently, companies can choose to buy back security tokens from the market for numerous reasons, such as to meet stock option obligations, improve financial ratios, take advantage of an undervalued share price, increase ownership, and reduce dilution.

After burning the security tokens, it does not have market value and no longer represents a share of ownership in the issuing corporation.

Pausable

Pausable also known as Stock Halt, is a temporary halt in the trading of a security. Usually, the halt is imposed for regulatory reasons, the anticipation of significant news, or to correct a situation in which there are excess buy or sell orders for a specific security.

The advantage of Pausable is to provide the entire market participant to be aware of some vital information about a stock or security. Also, to eradicate any kind of illegal practice of arbitrage options. This will protect investors from suffering substantial monetary losses.

Identity Management

For participating in any STO, users need to have an account at LCX. Users also have to pass KYC checks. When the user is fully verified, then we create an identity of the user on the Blockchain, which we call an Identity Registry.

Identity Registry is a single smart contract that we call is the Hub of Identities. The responsibility of this Registry is to define and enforce the rules of storing the Security tokens. The Identity of the user will be a unique identification number on the LCX platform, and on the blockchain, it will be represented as a Hash of that unique identification number.

This unique Identification number will map to all the core information relevant to the Identity, namely White Listed Addresses set, Recovery address, etc.

```
struct Identity {
    address RecoveryAddress;
    Addresses Whitelist Addresses;
    bytes32 data;

}
```

- Whitelisted Addresses are User's blockchain addresses who are KYC Approved by our Compliance Check on LCX Platform. Users can hold **Security tokens** only on a whitelisted address.
- Data field in the Identity stores the important information about which is relevant for Compliance check on the blockchain. We don't store any personal information of the users on the blockchain.
- In the event of irrecoverable loss of control of an Identity, Destruction is a contingency measure to permanently disable the Identity. It removes all Whitelisted Addresses, and data while preserving the Identity. Evidence of the existence of the Identity persists, while control over the Identity is nullified.

Compliance and KYC

Know Your Customer (KYC) procedures are a critical function to assess customer risk and a legal requirement to comply with Anti-Money Laundering (AML) laws. Effective KYC involves knowing a customer's identity, their financial activities, and the risk they pose.

LCX has a very advanced KYC system which is passed by the Liechtenstein Government. The user has to pass all compliance checks on the LCX platform then they will be able to participate in any Security token offering.

After fully verified, users can Whitelist their blockchain addresses. Where they can hold the Security token.

Wallet Whitelisting

Wallet Whitelisting is a process where a user can provide their blockchain address on the LCX platform and our Whitelisting Agent from the compliance team can whitelist your address on the blockchain.

The whitelisting process will be a Transaction on the Blockchain and cost us Fee. So for every user, one whitelisting will be free for everyone. But if a user wants to hold security tokens on more addresses then they have to pay fees in LCX for Whitelisting more addresses on the blockchain.

In this process, your address will also be added to the Identity Contract in a Whitelisted address.

In the Blockchain, you can know whether any wallet is whitelisted or not. We don't store any other information about the user in this process.

Token Holder Registry

The Real-time Cap table shows the current ownership, or token holding addresses, of security tokens. Cap table can be printed for the current moment or any moment in the past.

We are maintaining the real-time cap table on the blockchain with the help of Identity Contract and Whitelisted Addresses. And you can get the real-time cap table of any User with its unique identification number.

Ethereum Liechtenstein Protocol Token (ELP Token)

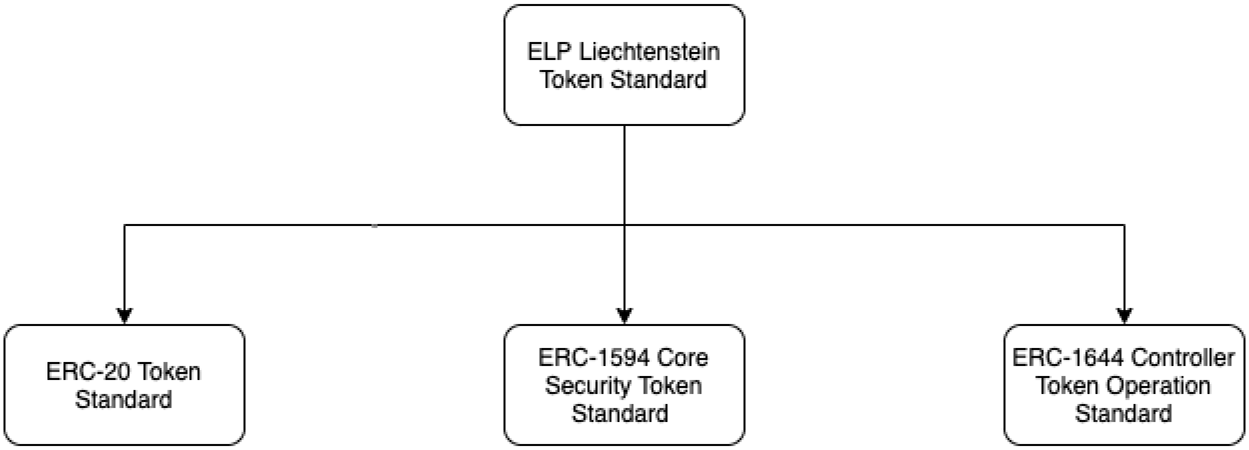
ELP Liechtenstein Token Standard
ERC-20 Token Standard
ERC-1594 Core Security Token Standard
ERC-1644 Controller Token Operation Standard

Figure - 2

The above figure i.e., *Figure-2,* explains the structure of the ELP Token Standard. It consists of the main three standards,

1. Token Standard (ERC-20)
2. Core Security Token Standard (ERC-1594)
3. Controller Token Operation Standard (ERC-1644)

ELP Token Standard

The Liechtenstein Token (ELP Token) is an authorized ERC-20 smart contract that can represent ownership of securities.

The following standard allows for the implementation of a standard API for tokens within smart contracts. This standard provides basic functionality to transfer tokens, as well as allow tokens to be approved so they can be spent by another on-chain third party. It is compatible with all existing wallets and exchanges that support the ERC-20 token standard.

This set of interfaces, contracts, and utilities are all related to the ERC20 Token Standard. There a few core contracts that implement the behavior specified in the EIP:

IERC-20

The interface of the ERC20 standard as defined in the EIP.

```
interface IERC20 {
    function totalSupply() external view returns (uint256);
    function balanceOf(address account) external view returns (uint256);
    function transfer(address recipient, uint256 amount) external returns (bool);
    function allowance(address owner, address spender) external view returns (uint256);
    function approve(address spender, uint256 amount) external returns (bool);
    function transferFrom(address sender, address recipient, uint256 amount) external returns  (bool);
    event Transfer(address indexed from, address indexed to, uint256 value);
    event Approval(address indexed owner, address indexed spender, uint256 value);
}
```

IERC20 defines function signatures without specifying behavior; the function names, inputs, and outputs, but no process. ERC20 inherits this Interface and is required to implement all the functions described or else the contract will not deploy.

ERC-20

One of the most significant Ethereum tokens is known as ERC-20. ERC-20 has emerged as the technical standard; it is used for all smart contracts on the Ethereum blockchain for token implementation and provides a list of rules that all Ethereum-based tokens must follow.

ERC-20 is similar, in some respects, to bitcoin, Litecoin, and any other cryptocurrency; ERC-20 tokens are blockchain-based assets that have value and can be sent and received. The primary difference is that instead of running on their own blockchain, ERC-20 tokens are issued on the Ethereum network.

The ERC-20 smart contract implements all the necessary functions needed to create a standard token, as seen in Appendix A (ERC-20 Functions).

ERC20 Detailed

It is used to initialize the name, symbol, and decimals for the token, but they aren’t necessary if your project doesn’t require a named ERC20 token.

Additionally, there are multiple custom extensions, including:

Burnable

It allows token holders to destroy both their own tokens and those that they have an allowance for, in a way that can be recognized off-chain (via event analysis).

Pausable

ERC20 with pausable transfers and allowances.

Useful if you want to stop trades until the end of a crowd-sale, or have an emergency switch for freezing all token transfers in the event of a large bug.

Finally, there are some utilities to interact with ERC20 contracts in various ways.

SafeERC20

Wrappers around ERC20 operations that throw on failure (when the token contract returns false). Tokens that return no value (and instead revert or throw on failure) are also supported, non-reverting calls are assumed to be successful. To use this library you can add a using SafeERC20 for IERC20; statement to your contract, which allows you to call the safe operations as a tokensafe transfer(…), etc.

Time Locking

A token holder contract that will allow a beneficiary to extract the tokens after the given release time. This is useful for simple vesting schedules like "advisors get all of their tokens after 1 year".

Core Security Token Standard

Transfers of securities can fail for a variety of reasons in contrast to utility tokens which generally only require the sender to have a sufficient balance.

These conditions could be related to the metadata of the securities being transferred (i.e. whether they are subject to a lock-up period), the identity of the sender and receiver of the securities (i.e. whether they have been through a KYC process, whether they are accredited or an affiliate of the issuer) or for reasons unrelated to the specific transfer but instead set at the token level (i.e. the token contract enforces a maximum number of investors or a cap on the percentage held by any single investor).

For ERC-20 tokens, the balance and allowance functions provide a way to check that a transfer is likely to succeed before executing the transfer, which can be executed both on and off-chain.

For tokens representing securities the standard introduces a function canTransfer / canTransferByPartition which provides a more general-purpose way to achieve this when the reasons for failure are more complex; and a function of the whole transfer (i.e. includes any data sent with the transfer and the receiver of the securities).

In order to provide a richer result than just true or false, a byte return code is returned. This allows us to give a reason for why the transfer failed, or at least which category of the reason the failure was in. The ability to query documents and the expected success of a transfer is included in the Security Token section.

In order to support off-chain data inputs to transfer functions, transfer functions are extended to transferWithData / transferFromWithData which can optionally take an additional bytes _data parameter.

```
interface IERC1594 is IERC20 {

    function transferWithData(address _to, uint256 _value, bytes _data) external;
    function transferFromWithData(address _from, address _to, uint256 _value, bytes _data) external;
    function isIssuable() external view returns (bool);
    function issue(address _tokenHolder, uint256 _value, bytes _data) external;
    function redeem(uint256 _value, bytes _data) external;
    function redeemFrom(address _tokenHolder, uint256 _value, bytes _data) external;
    function canTransfer(address _to, uint256 _value, bytes _data) external view returns (bool, byte,
bytes32);
    function canTransferFrom(address _from, address _to, uint256 _value, bytes _data) external view
returns (bool, byte, bytes32);
    event Issued(address indexed _operator, address indexed _to, uint256 _value, bytes _data);
    event Redeemed(address indexed _operator, address indexed _from, uint256 _value, bytes _data);

}
```

Whitelisting and Restricted Transfers

Transfers of securities may fail for a number of reasons, for example relating to:

- the identity of the sender or receiver of the tokens
- limits placed on the specific tokens being transferred (i.e. lockups on certain quantities of the token)
- limits related to the overall state of the token (i.e. total number of investors),

The standard provides an on-chain function to determine whether a transfer will succeed, and return details indicating the reason if the transfer is not valid.

These rules can either be defined using smart contracts and on-chain data or rely on _data passed as part of the transferWithData function which could represent authorization for the transfer (e.g. a signed message by a transfer agent attesting to the validity of this specific transfer).

If bytes _data is empty, then this corresponds to a check on whether a transfer (or transferFrom) request will succeed, if bytes _data is populated, then this corresponds to a check on transferWithData (or transferFromWithData) will succeed.

canTransfer assumes the sender of tokens is msg.sender and will be executed via **transfer** or **transferWithData** whereas **canTransferFrom** allows the specification of the sender of tokens and that the transfer will be executed via **transferFrom** or **transferFromWithData**.

Token Issuance

isIssuable: A security token issuer can specify that issuance has finished for the token (i.e. no new tokens can be minted or issued).

issue: This function must be called to increase the total supply. The bytes _data parameter can be used to inject off-chain data (e.g. signed data) to authorize or authenticate the issuance and receiver of issued tokens.

When called, this function MUST emit the Issued event.

Token Redemption

redeem: Allows a token holder to redeem tokens. The redeemed tokens must be subtracted from the total supply and the balance of the token holder. The token redemption should act like sending tokens and be subject to the same conditions.

When called, this function MUST emit the Issued event.

redeemFrom: This is the analogy to the redeem function, msg.sender MUST have a sufficient allowance set and this allowance must be debited by the _value.

Controller Token Operation Standard

Accelerate the issuance and management of securities on the Ethereum blockchain by specifying a set of standard interfaces through which security tokens can be operated on and interrogated by all relevant parties.

Since security tokens are subject to regulatory and legal oversight (the details of which will vary depending on the jurisdiction, regulatory framework, and underlying asset) in many instances the issuer (or a party delegated to by the issuer acting as a controller, e.g. a regulator or transfer agent) will need to retain the ability to force transfer tokens between addresses.

These controller transfers should be transparent (emit events that flag this as a forced transfer) and the token contract itself should be explicit as to whether or not this is possible.

Examples of where this may be needed is to reverse fraudulent transactions, resolve lost private keys and respond to a court order.

controllerTransfer

This function allows an authorized address to transfer tokens between any two token holders. The transfer must still respect the balances of the token holders (so the transfer must be for at most **balanceOf(_from) tokens**) and potentially also need to respect other transfer restrictions.

controllerTransfer MUST emit a ControllerTransfer event.

controllerRedeem

This function allows an authorized address to redeem tokens for any token holder. The redemption must still respect the balances of the token holder (so the redemption must be for at most balanceOf(_tokenHolder) tokens) and potentially also need to respect other transfer restrictions. controllerTransfer MUST emit a ControllerRedemption event.

```
interface IERC1644 is IERC20 {
    // Controller Operation
    function isControllable() external view returns (bool);
    function controllerTransfer(address _from, address _to, uint256 _value, bytes _data, bytes
_operatorData) external;
    function controllerRedeem(address _tokenHolder, uint256 _value, bytes _data, bytes _operatorData)
external;

    // Controller Events
    event ControllerTransfer(
        address _controller,
        address indexed _from,
        address indexed _to,
        uint256 _value,
        bytes _data,
        bytes _operatorData
    );
    event ControllerRedemption(
        address _controller,
        address indexed _tokenHolder,
        uint256 _value,
        bytes _data,
        bytes _operatorData
    );
}
```

LCX Token

The LCX Token ($LCX) is the fuel of the LCX.com platform and LCX Cryptocurrency Exchange.

LCX Token works as a long-term sustainable incentive mechanism to motivate various stakeholders to participate in the ecosystem.

The LCX Token is also a key part of the Liechtenstein Protocol as the utility token will be used for whitelisting services, issuing and other functionalities.

Conclusion

In summary, The Liechtenstein Protocol enables key elements for digital securities:

- **Compliance:** Fulfilling the compliance, regulatory and technology requirements to enable security token offerings at LCX platform, partners and other third parties.
- **Tokenization:** Creating a digital representative of financial instruments on the blockchain.
- **On-Chain Asset Management:** Management of the tokenized digital asset on token level and on the blockchain. The functionalities of this feature are inclusive of Issuance, timelocking, burning and transaction monitoring.
- **On-Chain Identity:** Single KYC and identity management on the blockchain.
- **Real-Time Cap-Table:** Enabling a real-time Token Holder Registry listing all holders of the digital security.

The Liechtenstein Protocol is the first step to create a decentralized compliance protocol for security-tokens. It offers solutions to current challenges blocking the growth of digital securities. This new security token standard solves how private securities can be issued, transferred and traded in a compliant manner.

Initiator

The Liechtenstein Protocol is initiated and developed by LCX AG, Liechtenstein Cryptoassets Exchange, headquartered in Vaduz, Liechtenstein.

www.LCX.com

References and Additional Information

- Security token technical design overview: https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781838551063/5/ch05lvl1sec33/security-token-technical-design-overview
- The Security Token Standard: https://hackernoon.com/the-security-token-standard-bc07409947ae
- T-REX (Token for Regulated EXchanges): https://tokeny.com/wp-content/uploads/2018/12/t-rex-whitepaper.pdf
- Security Token Standard: https://thesecuritytokenstandard.org/
- OECD Report -The Tokenisation of Assets and Potential Implications for Financial Markets: http://www.oecd.org/finance/The-Tokenisation-of-Assets-and-Potential-Implications-for-Financial-Markets.pdf
- ERC1400 Security Token Standard: https://polymath.network/erc-1400

Disclaimer

This white paper ("White Paper") is meant to describe the currently status and further anticipated plans of LCX AG ("LCX") of a new blockchain token protocol ("Liechtenstein Protocol") that defines a mechanism in which security-tokens can be compliantly transferred on the blockchain (the "LP Token"). Nothing in this document should be treated or read as a guarantee or promise of how the Liechtenstein Protocol or LP Token will develop or of the utility or value of the Liechtenstein Protocol or the LP Token. This White Paper outlines LCX's current plans, which could change at its discretion, and the success of which will depend on many factors outside LCX's control, including market-based factors and factors within the data and cryptocurrency industries, among others. Any statements about future events are based solely on LCX's analysis of the issues described in this document. That analysis may prove to be incorrect.

This document does not constitute an offer or sale of the Liechtenstein Protocol or LP Tokens or any other mechanism for purchasing the LP or LCX Tokens.