

XENFT公司

原文： XENFT

# 1.XEN Torrent协议

原文： 1. XEN Torrent Protocol

Fair Crypto Foundation Litepaper v0.3@Jack Levin先生@lbelyaev

原文： Fair Crypto Foundation Litepaper v0.3 @MrJackLevin @lbelyaev

# XENFT-XEN洪流（XENT）

原文： XENFT - XEN Torrent (XENT)

XEN加密

原文： XEN Crypto

XEN Crypto（XEN）是与EVM兼容的ERC-20令牌。XEN Crypto由Jack Levin设计，并于2022年由Fair Crypto Foundation发起，为创建以第一原则为中心的加密工具生态系统奠定了基础，该生态系统专为全球大众采用而设计。

原文： XEN Crypto (XEN) is an EVM compatible ERC-20 token. Devised by Jack Levin and launched by the Fair Crypto Foundation in 2022, XEN Crypto laid the groundwork to create an ecosystem of crypto instruments centered around the First Principles and designed for mass world adoption.

# XENFT公司

原文： XENFT

如上所述，XEN加密被实现为符合ERC-20标准的可移动令牌（FT）。它的主要目标是作为一种交流媒介。人们可以使用快速增长的合作伙伴生态系统造币、购买、入股和出售XEN。

原文： As mentioned above, XEN Crypto is implemented as a Fungible Token (FT) adhering to the ERC-20 standard. Its primary goal is to function as a medium of exchange. People can mint, buy, stake and sell XEN using the fast growing partner’s ecosystem.

随着XENFT的推出，公平加密基金会向XEN的使用和大规模采用迈出了新的一步。XENFT是XEN NFT的派生，是一种不可替代的令牌，通过铸造以及实现和使用防烧毁协议（XEN ERC-20智能合约本机）与XEN ERC20令牌互连。

原文： With the launch of XENFT, Fair Crypto Foundation takes a new step towards XEN use and mass adoption. XENFT, is a derivation of XEN NFT, is a non-fungible token that interconnects with XEN ERC20 token by virtue of minting as well as the implementation and use of the Proof of Burn protocol (native to XEN ERC-20 Smart Contract).

本文是XENFT系列论文的第一部分，涵盖XEN Torrent项目。因此，下文中对XENFT的任何提及都是指XEN Torrent智能合约所代表的XENFT。

原文： This paper is the first part of a series of litepapers dedicated to XENFT and covers XEN Torrent project. Thus any reference to XENFT in the text below refers specifically to XENFT represented by XEN Torrent Smart Contract.

## XEN洪流

原文： XEN Torrent

XEN Torrent是一种多面仪器，由ERC-721兼容合同运行。它可以部署在XEN Crypto合同已经运行的任何EVM兼容网络上。

原文： XEN Torrent is a multi-faceted instrument which is run by an ERC-721 compatible contract. It can be deployed on any EVM compatible network where XEN Crypto contract has already been operational.

与XEN一样，XEN Torrent坚持加密的第一原则：

原文： Same as XEN, XEN Torrent adheres to the first principles of crypto:

* 无预造币厂；

原文： No pre-mint;

* 没有白名单、黑名单或任何特殊分配；

原文： No whitelists, blacklists or any special allocations;

* 不变的合同；

原文： Immutable contract;

* 无管理（控制）键；

原文： No admin (control) keys;

### 公用事业

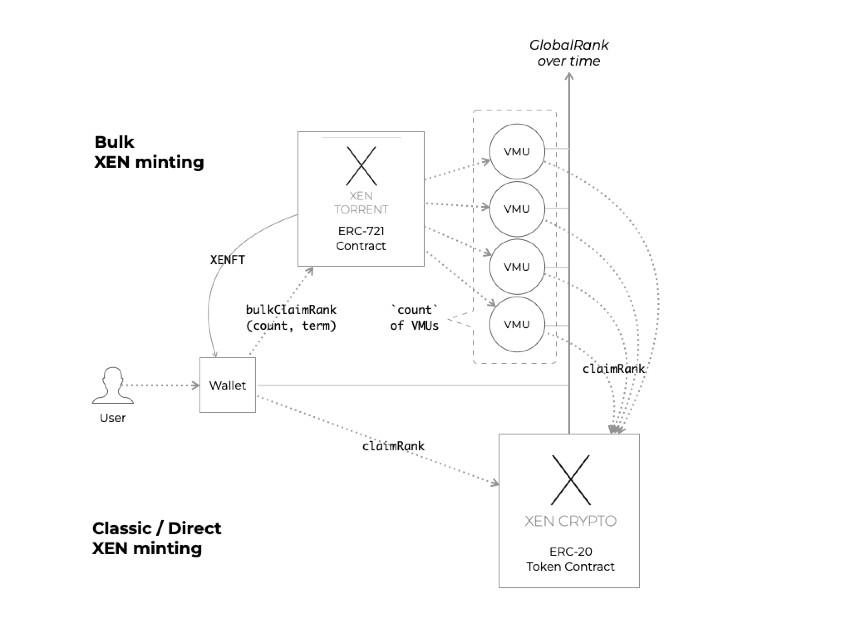
原文： Utility

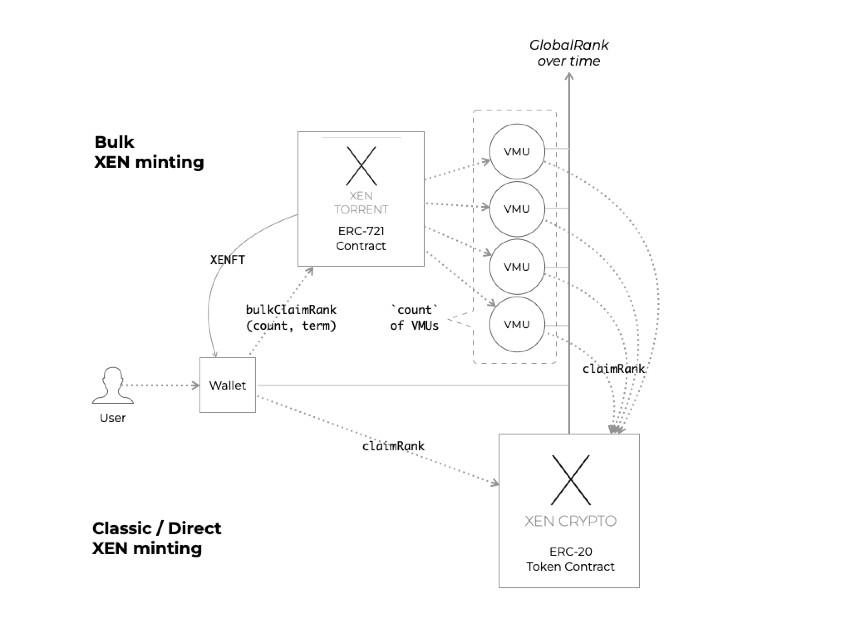
在幕后，XEN Torrent自动化了一系列链上交易的执行，通过虚拟化以太坊地址来最大化XEN（ERC-20）的造币量，该地址用于声明cRank（又名虚拟造币单位或VMU），由用户通过XEN Torent智能合约控制。

原文： Under the hood, XEN Torrent automates the execution of a series of on-chain transactions, maximizing XEN (ERC-20) mint by virtualizing Ethereum addresses used to claim cRanks (aka Virtual Minting Units, or VMUs), controlled by a user via XEN Torrent smart contract.

与XEN Crypto类似，它有两个阶段的操作（Claim Rank和Claim Mint Reward），XEN Torrent也有两个步骤。

原文： Similar to XEN Crypto, which has a two phase operation (Claim Rank and Claim Mint Reward), XEN Torrent also has two phases.

在初始阶段，将创建一组VMU，其数量由用户设置的“计数”参数控制。创建后，每个VMU使用ClaimRank函数调用原始XEN Crypto智能合约，启动XEN mintingClaimRank的Term参数也由用户设置。简而言之

原文： During the initial phase, a set of VMUs is created, whose number is controlled by the ‘count’ parameter set by a user. Upon creation, each VMU calls the original XEN Crypto smart contract with ClaimRank function, initiating XEN minting. ‘Term’ parameter for ClaimRank is also set by the user. In a nutshell,

第3页，共17页

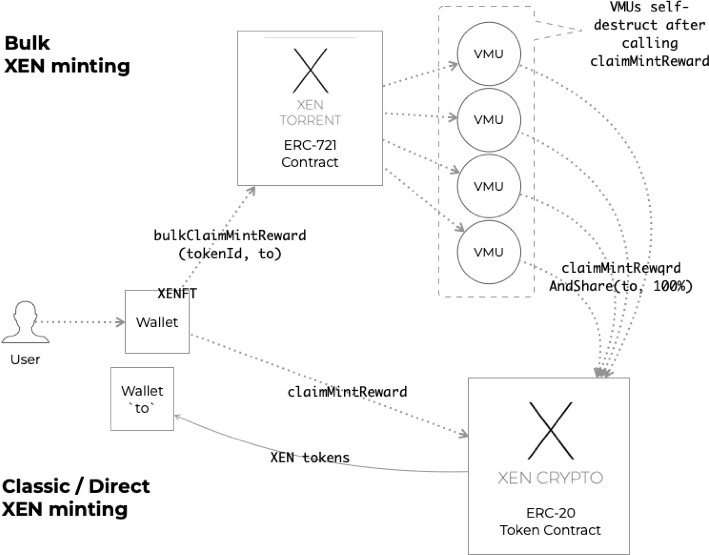
原文： 3 of 17

XEN Torrent允许使用单个事务从单个用户控制的地址开始并行铸造XEN。

原文： XEN Torrent allows to start the parallel minting of XEN from the single user-controlled address, using a single transaction.

当XEN Torrent协议控制的XEN铸造达到成熟期（自原始阶段发生以来的“期限”天数）时，用户可以使用第二个也是最后一个阶段。一旦达到成熟度，XEN Torrent Protocol用户可以通过受控VMU（在初始阶段创建）执行批量ClaimMintReward操作。然后，在此操作期间生成的XEN令牌将被传输到用户指定的地址（可以是用户的原始地址，也可以是网络上的任何其他地址）。

原文： Second and final phase is available for a user when XEN minting controlled by XEN Torrent Protocol reaches maturity (the ‘term’ amount of days passes since the original phase took place). Once the maturity is reached, XEN Torrent Protocol user can perform a bulk ClaimMintReward operation via controlled VMUs (created on the initial phase). XEN tokens minted during this operation are then transferred to a user-designated address (which could be user’s original address, or any other address on the network).



### 资产

原文： Asset

XEN Torrent发起批量铸造XEN Crypto的每个用户操作都会发布一个不可替换的令牌（XENFT/ERC-721），该令牌将被传送给用户。每个XENFT都是唯一的，不能与任何其他XENFTT相互互换（因此不可替换）。

原文： Each user operation with XEN Torrent initiating bulk minting of XEN Crypto issues a non-fungible token (XENFT/ERC-721) which is transferred to the user. Each XENFT is unique and is not mutually-interchangeable with any other XENFT (thus non-fungible).

任何用户（通过钱包地址或智能合约表示）都可以拥有无限数量的XENFT。每个新发布的XENFT代表对批量XEN铸造操作的访问控制权限。为了在到期后申领XEN代币，用户必须拥有XENFT。

原文： Any user (represented either via a wallet address or a smart contract) can own an unlimited number of XENFTs. Each newly issued XENFT represents an access control rights to the bulk XEN minting operation. In order to claim XEN tokens once the maturity is reached, the user has to be in possession of the XENFT.

由于XEN Torrent代币符合ERC-721 NFT标准，因此可转让。这意味着一个用户可以将其拥有的XENFT转让给另一个用户（无论转让是由于出售或交换还是免费进行，都不在本文和XEN Torrent智能合约的范围内）。

原文： Since XEN Torrent tokens adhere to ERC-721 NFT standard, they are transferrable. This means that one user can transfer a XENFT owned by them to another user (whether transfer happens as a result of a sale or swap or free of charge is out of the scope of this paper and XEN Torrent smart contract).

如上所述，XEN Torrent XENFT作为一种访问令牌，用于请求批量XEN铸造的权利。这是一种经典的不记名代币，这意味着任何拥有（赠送）XEN Torrent代币的人都可以申领XEN ERC-20铸币的收益。

原文： As mentioned above, XEN Torrent XENFT serves as an access token to claim rights to the bulk XEN minting. It’s a classic bearer token, which means that whoever owns (presents) XEN Torrent Token can claim the proceeds of XEN ERC-20 mint.

### XENFT属性

原文： XENFT Properties

每个已发布的XENFT将具有以下属性，并将其编码和存储在XEN Torrent智能合约中：

原文： Each of issued XENFTs will have the following properties encoded and stored in the XEN Torrent smart contract:

* XEN加密铸造相关

原文： XEN Crypto minting related

* + 期限（天）

原文： Term (days)

* + 到期时间戳

原文： Maturity Timestamp

* + cRank（与第一个VMU或批处理的开始有关）

原文： cRank (related to the first VMU, or the start of the batch)

* + AMP（与第一个VMU或批次开始有关）

原文： AMP (related to the first VMU, or the start of the batch)

* + EAA（与第一个VMU或批次开始有关）

原文： EAA (related to the first VMU, or the start of the batch)

* 特定于XEN Torrent

原文： XEN Torrent specific

* + VMU计数

原文： Count of VMUs

* + 类别（见下文）

原文： Category (see below)

* + XEN燃烧量（见下文）

原文： Amount of XEN burned (see below)

* + 已赎回（或未赎回）

原文： Redeemed (or not)

XENFT中捕获的所有属性都是不变的，除了最后一个属性。”“Redemed”属性是一个布尔值，在XENFT铸造时设置为“false”。一旦用户执行了第二阶段并声称

原文： All the properties captured in the XENFT are immutable, except for the last one. ‘Redeemed’ prop is a boolean value and is set to ‘false’ upon XENFT minting. Once the user has performed the second phase and claimed the

铸造XEN ERC20代币，XENFT被称为已赎回，“已赎回”属性设置为True。

原文： minted XEN ERC20 tokens, XENFT is said to be redeemed and the ‘redeemed’ property is set to True.

### 申请cRank和铸币的重要注意事项

原文： Important notes on claiming cRank and Minting

* 一旦XENFT发布给用户，XEN Torrent智能合约对代币没有控制权或产权负担。XEN Torrent智能合约在没有当前XENFT所有者明确指示的情况下，无法逆转代币销售或转让、冻结代币或自行对其执行任何操作。

原文： Once XENFT is issued to a user, XEN Torrent smart contract has no control or encumbrance over the token. XEN Torrent smart contract cannot reverse token sale or transfer, freeze the token or perform any operation on it by itself, without explicit instruction from the current XENFT owner.

* 通过XEN Torrent智能合约启动的XEN铸造操作由XEN Crypto ERC-20智能合约运行；因此，申请等级和申请铸币奖励的规则相同。通过XEN Torrent发起的XEN铸造没有特殊待遇，它与通过XEN Crypto直接操作共享相同的全球排名计数器。同样适用于铸币厂索赔/取款罚款；鼓励用户注意到期日和7天提款时间窗口。

原文： XEN minting operation initiated via XEN Torrent smart contract is run by XEN Crypto ERC-20 smart contract; whereby the same rules of claiming rank and claiming mint rewards apply. XEN minting initiated via XEN Torrent has no special treatment, it shares the same Global Rank counter as direct operations via XEN Crypto. Same is true for mint claim/withdrawal penalties; users are encouraged to be mindful of the maturity date and the 7 day withdrawal time window.

## XEN Torrent XENFT的分类

原文： Categories of XEN Torrent XENFTs

XEN Torrent XENFT有三种不同的口味和类别：

原文： XEN Torrent XENFTs come in 3 different flavors and categories:

* 稀有的

原文： Rare

* 有限的

原文： Limited

* 普通或普通

原文： Common or Ordinary

### 稀有的

原文： Rare

稀有类别是XEN Torrent层次结构中的顶级。它的发行仅限于智能合约控制的10000个NFT代币的总量，并且是不可变的。为了铸造稀有XENFT，必须满足以下标准：

原文： Rare category is the top tier in XEN Torrent hierarchy. Its issuance is limited to the total amount of 10,000 NFT tokens controlled by the smart contract and is immutable. In order to mint a Rare XENFT, the following criteria must be met:

* 发行的稀有XEN Torrent XENFT的总供应量小于或等于10000，

原文： Total Supply of issued Rare XEN Torrent XENFTs is less or equal to 10,000,

* 散装XEN薄荷中VMU的计数为100或更多，

原文： Count of VMUs in the bulk XEN mint is 100 or more,

* 用户拥有并愿意燃烧XEN代币，以获得铸造稀有XENFT的特权

原文： User is in possession and is willing to burn XEN tokens for the privilege to mint one of the Rare XENFTs

为稀有XENFT烧毁的XEN代币数量决定了稀有等级（稀有类别内），每个类别都有自己的发行代币限制：

原文： Amount of XEN tokens to be burned for the Rare XENFT determines the rarity class (inside the Rare category), each of which has its own limits of issued tokens:

* 1类：100个稀有XENFT（令牌ID 1…100）

原文： Class 1: 100 of Rare XENFTs (token IDs 1…100)

* 第2类：900个稀有XENFT（令牌ID 101…1000）

原文： Class 2: 900 of Rare XENFTs (token IDs 101…1,000)

* 类别3:2000稀有XENFT（令牌ID 1001…3000）

原文： Class 3: 2,000 of Rare XENFTs (token IDs 1,001…3,000)

* 第4类：3000个稀有XENFT（令牌ID 3001…6000）

原文： Class 4: 3,000 of Rare XENFTs (token IDs 3,001…6,000)

* 第5类：4000个稀有XENFT（令牌ID 6001…10000）

原文： Class 5: 4,000 of Rare XENFTs (token IDs 6,001…10,000)

在XEN Torrent合同部署日期之前，将定义制造每个类别的一个XENFT所需的XEN的具体数量，并且对于部署XEN Torrent的每个网络可能不同。然而，经验法则是，与较高阶层相比，每个较低阶层需要的氙气燃烧量将有所下降。

原文： Specific amounts of XEN which is required to mint one of the XENFTs of each class is to be defined until the XEN Torrent contract deployment date and could be different for each network where XEN Torrent is deployed. However the rule of thumb is that each lower class will require a declining scale of XEN burn compared to the higher one.

### 有限的

原文： Limited

与稀有类别不同，有限类别不受发行代币数量的限制，而是受时间的限制。根据以下标准，将向任何数量的合格用户发放有限的XENFT：

原文： Limited category, unlike the Rare category, is not limited by the number of tokens issued but rather is limited by time. Limited XENFTs will be issued to any number of qualified users according to following the criteria:

* 有限XENFT发布时间为自XEN Torrent智能合约部署（由“genesisTs”不可变变量捕获）起365天（31536000秒），

原文： Time of Limited XENFT issuance is 365 days (31,536,000 seconds) since the moment of XEN Torrent smart contract deployment (which is captured by the `genesisTs` immutable variable),

* 散装XEN薄荷中VMU的计数为100或更多，

原文： Count of VMUs in the bulk XEN mint is 100 or more,

* 用户拥有并愿意燃烧XEN代币，以获得铸造有限XENFT之一的特权。

原文： User is in possession and is willing to burn XEN tokens for the privilege to mint one of the Limited XENFTs.

就像稀有XENFT类别一样，在XEN Torrent合同部署日期之前，需要定义制造一个有限XENFTs所需的XEN烧毁的具体数量，并且对于部署XEN Torrent的每个网络可能会有所不同。然而，该金额将低于最低级别稀有XENFT类别所需的金额。

原文： Just like with the Rare XENFT category, the specific amount of XEN burn which is required to mint one of the Limited XENFTs is to be defined until the XEN Torrent contract deployment date and could be different for each network where XEN Torrent is deployed. However this amount will be less than amount required by the lowest class of Rare XENFT category.

每个用户可以持有的有限XENFT数量没有限制。

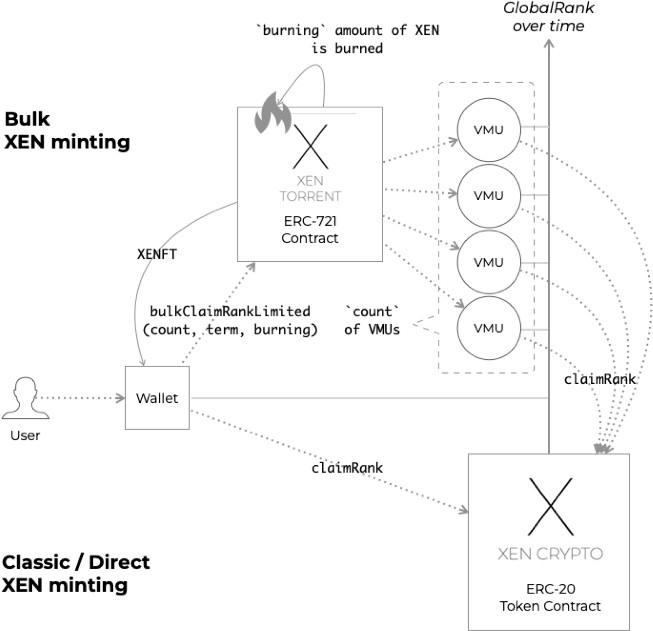
原文： There’s no limit to how many Limited XENFTs each of users can hold.

有限XENFT的令牌ID与普通/普通XENFTs共享相同的计数器（令牌ID从10001开始）。

原文： Token IDs for Limited XENFTs share the same counter with Ordinary / Common XENFTs (token IDs start at 10,001).

铸造稀有或有限类型的XENFT与燃烧XEN的差异如下图所示。

原文： Difference of minting Rare or Limited categories of XENFTs with regards to burning XEN is shown in the diagram below.



### 普通/普通

原文： Ordinary / Common

任何人都可以在任何时间、任何次数免费铸造此类XENFT。用户的唯一承诺（适用于所有XENFT类别）是支付初始XENFT薄荷交易的网络燃气费。不需要XEN（ERC-20）燃烧。

原文： XENFTs of this category can be minted free by anyone, at any time, any number of times. The only commitment from a user (universal for all XENFT categories) is to pay the network gas fee for initial XENFT mint transaction. XEN (ERC-20) burn isn’t required.

普通XENFT可以铸造为低至1个VMU，期限低至1天。

原文： Common XENFTs can be minted to represent as low as 1 VMU with the term as low as 1 day.

为了使普通XENFT“不那么常见”，并为所有XENFTs所有者引入额外的博弈论精神，XEN Torrent普通代币也被分为不同的类别。在这种情况下，限定参数是XENFT的“功率”，计算如下：

原文： In order to make Common XENFTs ‘less common’ and introduce additional game theory spirit for all XENFTs owners, XEN Torrent Common tokens are also divided into classes. In this case, qualifying parameter is XENFT’s `power` which is computed as:

#### P𝚘𝚠𝚎𝚛 = | 𝚅MU| \* 𝚝𝚎𝚛𝚖,

或者，简单地说，它是启动的VMU数量乘以XEN铸造期限（以天表示）。

原文： Or, in plain terms, it’s the count of launched VMUs times XEN minting term (expressed in days).

这种合成属性的选择取决于它作为未来XEN奖励价值的最佳代理，并且可以用作两种不同XENFT之间的快速（尽管非常近似）比较因素。

原文： The choice of this synthetic property is determined by the fact that it serves as the best proxy to the value of future XEN rewards, and could be used as a quick (albeit very approximate) comparison factor between two different XENFTs.

普通XENFT的类别定义如下：

原文： Classes of Common XENFTs are defined as:

#### 𝙲𝚒 = ⌊P𝚘𝚠𝚎𝚛/ 𝟽𝟻𝟶𝟶

相应地（按如上所示计算的类别指数）：

原文： Accordingly (by class index computed as shown above):

* 0级（红宝石）：功率1…7500

原文： Class 0 (Ruby): Power 1…7,500

-1级（蛋白石）：功率7501…15000

原文： - Class 1 (Opal): Power 7,501…15,000

-2级（黄玉）：功率15001…22500

原文： - Class 2 (Topaz): Power 15,001…22,500

* 3级（翡翠）：功率22501…30000

原文： Class 3 (Emerald): Power 22,501…30,000

* 4级（海蓝）：功率30001…37500

原文： Class 4 (Aquamarine): Power 30,001…37,500

* 5级（蓝宝石）：功率37501…45000

原文： Class 5 (Sapphire): Power 37,501…45,000

* 6级（紫水晶）：功率45001…52500

原文： Class 6 (Amethyst): Power 45,001…52,500

* 7级（氙气）：电源52501…

原文： Class 7 (Xenturion): Power 52,501…

注意，普通XENFT的第7类在“功率”参数方面是无界的；任何功率高于52501的XENFT都将被归类为氙气。

原文： Note that Class 7 of Common XENFTs is unbounded with regards to the `power` parameter; any XENFT with power above 52,501 will be classified as Xenturion.

### XENFT元数据艺术

原文： XENFT Metadata Art

公平加密基金会相信，XENFT的价值将由其效用和稀有性（类别和类）属性决定。

原文： It is Fair Crypto Foundation’s belief that XENFT value will be determined by both its utility and its rarity (category and class) properties.

元数据艺术是XENFT价值的一种非常重要的表现形式，因此，所有令牌类别和类都将承载不同的艺术，捕捉XENFT价值的各个方面。

原文： Metadata art is a very important manifestation of the XENFT value, therefore all token categories and classes will bear distinct art capturing various aspects of the XENFT value.

（以下所有示例仅供说明；设计可能会在实际部署日前更改）。

原文： (All examples below are for illustrative purposes; designs could change before actual deployment day).

稀有XENFT封面示例。请注意不同的半透明颜色和表示稀有类别的盈阳标记。

原文： Example of a Rare XENFT cover. Note the distinct translucent colors and Ying-Yang mark denoting Rare category.

限量版XENFT封面艺术的例子。请注意不同的半透明颜色和表示限量版的烧焦标记。

原文： Example of a Limited XENFT cover art. Note the distinct translucent colors and burn mark denoting Limited category.

各种类型的普通XENFT封面艺术示例如下所示。注意不同的颜色/类名和表示公共类别的拾取轴标记。

原文： Example of various classes of Common XENFT cover art are shown below. Note the different colors / class names and Pick Axe mark denoting Common category.

### XENFT元数据属性的一般信息

原文： General info on XENFT metadata properties

XEN Torrent合同支持ERC-721扩展，允许任何人通过“tokenURI”方法查询XENFT元数据中的任何令牌ID。所述方法返回JSON格式的字符串，该字符串包含XENFT的各种属性，包括用于检索唯一XENFT图像的URL。

原文： XEN Torrent contract supports ERC-721 extension which allows anyone to query XENFT metadata for any token ID via `tokenURI` method. The said method returns a JSON-formatted string that contains various properties of XENFTs, including URL to retrieve unique XENFT image.

遵循去中心化原则，XEN代币合同正在生成所有元数据，包括链上唯一的元数据艺术图像。图像作为智能合约生成的数据URL base64编码字符串返回。

原文： Following the principles of decentralization, XEN Token contract is generating all metadata including unique metadata art image on-chain. Image is returned as data URL base64-encoded string generated by smart contract.

## XENFT和XEN加密

原文： XENFT and XEN Crypto

如上所述，XEN Torrent和XEN Crypto协同工作：

原文： As mentioned above, XEN Torrent and XEN Crypto work together in concert:

* XEN Crypto合约地址在部署时被捕获在XEN Torrent智能合约中，并且是不可变的；

原文： XEN Crypto contract address is captured in the XEN Torrent smart contract at the moment of its deployment and is immutable;

* 当XEN Torrent合同要求铸造稀有和有限类别的XENFT时，XEN Token合同通过作为XEN Crypto协议一部分的“烧毁证明”机制烧毁所有批准的XEN令牌；

原文： When required by XEN Torrent contract for minting XENFTs of Rare and Limited categories, XEN Token contract burns of all of the approved XEN tokens via `Proof-Of-Burn` mechanism which is a part of XEN Crypto protocol;

* XEN Torrent合约（通过VMU）使用XEN Crypto协议声明排名，然后造币XEN代币；

原文： XEN Torrent contract (via VMUs) uses XEN Crypto protocol to claim rank and then mint XEN tokens;

N、 B.XEN Crypto或XEN Torrent合同均未拥有其任何资产。所有XEN代币和XENFT始终归用户所有。

原文： N.B. Neither of XEN Crypto or XEN Torrent contracts ever own any of their assets. All XEN tokens and XENFTs are always owned by users.

## XENFT的二次销售

原文： Secondary sales of XENFT

如上所述，XEN Torrent实现了ERC-721令牌标准，该标准允许令牌（XENFT）在用户之间无任何限制地传输。

原文： As mentioned above, XEN Torrent implements ERC-721 token standard, which allows for tokens (XENFTs) to be transferred between users without any limitations.

导致XENFT所有权在账户之间转移的用户之间交易的性质远远超出了本白皮书的范围，但有几点需要牢记：

原文： The nature of deals between users that result in XENFT ownership being transferred between accounts is well outside the scope of this light paper, however there are several important things to keep in mind:

* XENFT所有权的任何转让均由卖方发起或买方发起，

原文： Any transfer of XENFT ownership is either seller-initiated or buyer-initiated,

* 卖家发起的转账（自有账户之间的转账、XENFT的“赠与”等可通过单笔交易完成——

原文： A seller-initiated transfer (transfer between owned accounts, ‘gifting’ of XENFTs, etc. could be accomplished via single transaction - a call to

`transferFrom（）`函数，其中当前所有者声明令牌ID和令牌受让方（“收件人”地址）。

原文： `transferFrom()` function, where current owner states token ID and the token transferee (`to` address).

* 买方发起的转让（通常发生在“出售”交易中，通常通过Opensea、Rarible等NFT市场）通常是一个两步过程：

原文： A buyer-initiated transfer (commonly occurring in as ‘sale’ transaction, typically via some sort of NFT marketplace like Opensea, Rarible, etc.) is typically a 2-step process:

* + 第一步是“批准”功能调用，将“授权书”授予第三方代理（通常是市场）进行未来转让的权利

原文： 1st step is `approve` function call, giving `power-of-attorney` right to a 3rd party agent (typically a marketplace) to make future transfer

* + 第二步是“transferFrom”事务调用，它列出

原文： 2nd step is the `transferFrom` transaction call which lists addresses of

`从“和”到“主题和令牌ID。注意：

原文： `from` and `to` subjects and the token ID. N.B.:

* + `如果由第三方发起，未经事先“批准”的transferFrom交易将失败！

原文： `transferFrom` transaction without prior `approve` transaction will fail if initiated by a third party!

* + 用户在发布“批准”交易时需要小心，因为他们基本上被授予了第三方命令其NFT的权利

原文： Users need to be careful when issuing an `approve` transaction since they are basically given away the right to a 3rd party to command their NFT(s)

* + 与委托书一样，“批准”权力可以在所有权转让之前重新修改

原文： Same as with power-of-attorney, `approve` powers could be reworked before the transfer of ownership occurred

* + 如果XENFT当前和/或新所有者不是钱包而是智能合约，则该合约应特别设计为能够持有ERC-721代币。如果这样的合同不支持它，向他们转让XENFT将失败！

原文： If XENFT current and/or new owner is not a wallet but rather a smart contract, such contract should be specially designed to be able to hold ERC-721 tokens. If such contracts don’t support it, transfer of an XENFT to them will fail!

# 技术细节

原文： Technical Details

## XEN Torrent合同结构

原文： XEN Torrent Contract Structure

XEN Torrent是一个与ERC-721兼容的智能合约，它基于广泛接受并经过良好测试的OpenZeppelin参考实现。

原文： XEN Torrent is a ERC-721 compatible smart contract which is based on widely accepted and well-tested OpenZeppelin reference implementation.

由于大量代码和EVM智能合约的24 Kb字节代码限制，XEN Torrent被分为不同的部分（库）：

原文： Due to a large amount of code and given 24 Kb byte code limit for EVM smart contracts, XEN Torrent is broken into separate parts (libraries):

### 薄荷信息

原文： MintInfo

MintInfo定义了一个结构，用于记录与单独批量造币操作（Torrent）相关的所有元素。为了节省汽油，MintInfo将7种不同的属性打包到一个uint256存储变量中。MintInfo库具有对单个记录进行编码和解码的方便方法，以及对记录的单独属性进行访问的访问器。

原文： MintInfo defines a structure for recording all elements related to a separate bulk mint operation (a Torrent). In the interest of saving gas, MintInfo packs 7 different properties into a single uint256 store variable. MintInfo library has convenience methods for encoding and decoding of a single record, as well as accessors for separate properties of a record.

### SVG公司

原文： SVG

SVG是一个库，负责生成对应于每个XENFT的唯一图像（示例如上文所示）。它定义了几种用于将参数传递给SVG的结构化类型（数据参数、颜色编码参数、渐变参数）。该库唯一可外部访问的方法是image（），它返回SVG图像的字节字符串。

原文： SVG is a library which is responsible for producing of a unique image corresponding to each XENFT (examples were shown above). It defines several structured types that are used to pass parameters to SVG (data params, Color encoding params, Gradient params). The only externally accessible method of this library is image(), which returns a byte string of an SVG image.

### 字符串数据

原文： StringData

StringData是一个小型库，用于存储和访问XEN引号（用于XENFT元数据艺术）和不同XENFT系列的名称。

原文： StringData is a small library which deals with storing and accessing of XEN quotes (used in XENFT metadata art) and names of different XENFT series.

### 日期时间

原文： DateTime

DateTime是一个库，用于将Unix历元时间戳（自1970年1月1日起经过的秒数）转换为人类可读的日期和时间字符串（UTC时区）。它基于BokkyPooBah的开源库

原文： DateTime is a library for converting of Unix epoch timestamps (number of seconds elapsed since Jan 1, 1970) to human readable Date and Time strings (in UTC timezone). It is based on an open-sourced library BokkyPooBah's

DateTime Library v1.01。DateTime库导出的主要函数是asString（unit256 ts），它返回所提供时间戳的字符串表示。

原文： DateTime Library v1.01. Main function exported by DateTime library is asString(unit256 ts) which returns string representation of a supplied timestamp.

元数据

原文： MetaData

元数据库包含生成XENFT元数据的主要构建块，包括图像构建和创建具有所有XENFT属性的JSON编码对象。它导出2个函数：svgData（…）和属性（uint256 count，uint256mintInfo）。

原文： MetaData library contains the main building blocks for producing XENFT metadata, including image building and creation of a JSON-encoded object with all XENFT properties. It exports 2 functions: svgData(…) and attributes(uint256 count, uint256 mintInfo).

### X校正

原文： XENTorrent

XENTorrent是主要的智能合约，它扩展了ERC-721样板，实现了特定的逻辑，并存储了与XEN Torrent协议相关的数据。

原文： XENTorrent is the main smart contract which extends ERC-721 boilerplate and implements specific logic and stores data related to XEN Torrent protocol.

XENTorrent的构造函数具有以下签名：

原文： Constructor of XENTorrent has the following signature:

构造函数（地址xenCrypto，uint256[]内存burnRates\_，uint256[]内存标记Limits\_）

原文： constructor(address xenCrypto\_, uint256[] memory burnRates\_, uint256[] memory tokenLimits\_)

它接受并存储XEN Torrent协议中使用的几个重要的不可变值：

原文： It accepts and stores several important immutable values used in XEN Torrent protocol:

* 原始XEN Crypto合同的地址（xenCrypto\_）

原文： address of original XEN Crypto contract (xenCrypto\_)

* 用于制造稀有和有限XEN Torrent NFT的XEN燃烧率参数（burnRates\_）

原文： XEN burn rate parameters used to mint Rare and Limited XEN Torrent NFTs (burnRates\_)

* 每个稀有和有限XEN洪流系列的极限（tokenLimits\_）

原文： Limits of each of Rare and Limited XEN Torrent series (tokenLimits\_)

XEN Torrent智能合约的公共接口由以下方法组成：

原文： Public interface of XEN Torrent smart contract consists of the following methods:

只读

原文： Read only

owner（）-返回XEN Torrent合同的部署者地址。

原文： owner() - returns the deployer address for the XEN Torrent contract.

用于在Opensea和其他市场上设置NFT集合参数。

原文： Used to set up NFT collections params on Opensea and other marketplaces.

ownedTokens（）-返回一个数组，其中包含当前用户（调用此方法的用户）的所有XEN Torrent令牌的令牌ID。

原文： ownedTokens() - returns an array containing tokenIds of all XEN Torrent tokens for the current user (who calls this method).

tokenURI（）-返回一个数据URL引用的JSON编码字符串，其中包含每个唯一XENFT所需的元数据对象。

原文： tokenURI() - returns a data URL- referred JSON-encoded string containing required metadata object for each unique XENFT.

写入（事务）

原文： Write (transactions)

bulkClaimRank（…）使用“count”VMU和“term”天开始批量XEN生成操作，并向调用用户发出XENFT。

原文： bulkClaimRank(…) starts bulk XEN minting operation with `count` VMUs and for `term` days and issues XENFT to the calling user.

bulkClaimRankLimited（…）使用“count”VMU和“term”天开始批量XEN造币操作，并向调用用户发出XENFT。用于请求有限XENFT中的稀有（通过指定要燃烧的XEN量的“燃烧”参数）。如上所述，此调用成功后将烧掉指定数量的XEN。

原文： bulkClaimRankLimited(…) starts bulk XEN minting operation with `count` VMUs and for `term` days and issues XENFT to the calling user. Is used to request a Rare of a Limited XENFT (via `burning` param that specifies amount of XEN to be burned). As mentioned above, the specified amount of XEN is burned on success of this call.

bulkClaimMintReward（…）终止当前的批量XEN造币操作，声明XEN令牌并销毁所有VMU。此方法的访问权限仅限于XENFT“tokenId”所有者。成功呼叫将XENFT的状态更改为已赎回=真。

原文： bulkClaimMintReward(…) terminates current bulk XEN minting operation, claims XEN tokens and destructs all VMUs. Access to this method is limited to XENFT `tokenId` owner. Successful call changes the state of XENFT to redeemed = true.

## VMU实施

原文： VMU Implementation

VMU（虚拟铸币单位）是一个链上钱包地址，由XEN Torrent合同创建。每个VMU本身都是一个智能合约，由用户通过XEN Torrent合约控制（因为智能合约没有私钥，因此无法自行签署交易）。

原文： VMU (Virtual Minting Unit) is an on-chain wallet address that is being created by the XEN Torrent contract. Each VMU is itself a smart contract which is controlled by a user via XEN Torrent contract (since smart contracts don’t have private keys and therefore cannot sign transactions by themselves).

VMU智能合约是根据“最小代理”模式（EIP-1167）创建的，该模式允许廉价克隆现有合约-在我们的案例中是XEN Torrent合约。为了安全起见，为了区分原始XEN Crypto及其克隆（VMU），原始合同地址通过构造函数中的不可变变量集记录在自身中。

原文： VMU smart contract is created according to a `Minimum Proxy` pattern (EIP-1167) which allows to cheaply clone an existing contract - in our case XEN Torrent Contract. For security, in order to distinguish between the original XEN Crypto and its clones (VMUs), original contract address is recorded in itself via immutable variable set in constructor.

每个VMU的外部接口包括以下方法：

原文： External interface of every VMU consists of the following methods:

callClaimRank（）调用ClaimMintReward（）

原文： callClaimRank() callClaimMintReward()

断电（）

原文： powerDown()

前2个方法对XEN Crypto合约进行代理调用，以相应地声明秩和声明薄荷。

原文： First 2 methods make proxy calls to XEN Crypto contract to claim rank and claim mint correspondingly.

最后一种方法使VMU收缩到自毁状态。

原文： The last method makes VMU contract to self-destruct.

所有这些方法都只能由原始的XEN Torrent智能合约调用，这允许用户有效地访问控制以开始批量造币并在到期时终止。

原文： All these methods are callable only by the original XEN Torrent smart contract, which allows effective access control for a user to start bulk mint and to terminate it at maturity.

## 燃烧XEN并防止再次进入

原文： Burning XEN and Preventing Reentrancy

如上所述，获得稀有或有限类别的XEN Torrent XENFT需要用户烧毁一定数量的XEN代币。为了实现原子化（烧毁XEN以换取铸造的XENFT），该交易需要XEN Torrent和XEN Crypto合同之间的两相通信。

原文： As mentioned above, getting Rare or Limited category of XEN Torrent XENFT requires user to burn a certain amount of XEN tokens. In order to be atomic (burned XEN in exchange for minted XENFT), this transaction requires a two phase communication between XEN Torrent and XEN Crypto contracts.

阶段1.旋转VMU和批量索赔等级。完成后，XEN Torrent调用XEN Crypto合约的burn（）方法，其中包含令牌所有者和令牌数量的详细信息。

原文： Phase 1. Spin VMUs and bulk claim rank. On completion, XEN Torrent calls burn() method of XEN Crypto contract with details of token owner and token quantity.

阶段2.一旦XEN令牌被XEN Crypto合约烧毁，它将通过onTokenBurned（）方法调用XEN Torrent合约。该方法完成原子事务，生成XENFT并将其MintInfo记录到合约存储中。

原文： Phase 2. Once XEN tokens are burned by XEN Crypto contract, it calls back XEN Torrent contract via onTokenBurned() method. This method completes the atomic transaction, mints XENFT and records its MintInfo into the contract storage.

为了在两个阶段之间保持状态，并防止重入攻击，使用了私有变量\_tokenId。它在第1阶段设置，在第2阶段清除。如果合约在\_tokenId为非零时收到对初始第1阶段方法的调用，或者如果在\_token Id为零时收到回调，则该事务将失败。

原文： In order to maintain state between the 2 phases, and also to prevent reentrancy attacks, a private variable \_tokenId is used. It’s set in phase 1 and cleared in phase 2. If the contract would receive either a call to initial phase 1 method while \_tokenId is non-zero, OR if a callback would be received with \_tokenId is zero, such transaction would fail.

## 燃烧证明协议

原文： Proof-of-Burn Protocol

与XEN Crypto相同，XEN Torrent支持防烧毁协议，该协议允许任何第三方智能合约烧毁XENFT-例如，以换取其他加密令牌。

原文： Same as XEN Crypto, XEN Torrent supports Proof-of-Burn protocol, which allows any 3rd party smart contract to burn XENFTs - for example in exchange for some other crypto token (or tokens).

燃烧证明协议由两部分组成：

原文： Proof-of-Burn protocol consists of 2 parts:

1. 主合约（令牌控制器）实现可由另一个合约（令牌代理）调用的方法burn（）。与上述烧录XEN的情况一样，调用此方法将启动“烧录证明”事务。

原文： Main contract (token controller) implements method burn() callable by another contract (token agent). As in the case with burning XEN described above, calling this method initiates Proof-of-Burn transaction.

1. 令牌代理合约实现IBurnRedeemable接口，该接口包括onTokenBurned（）方法和Redeemed事件。该合同还需要实施ERC-165标准，特别是它的supportsInterface（）方法，该方法应肯定地响应关于支持IBurnRedemable接口的请求。令牌控制器调用onTokenBurned（）方法，并在刻录过程完成后发出Redemed事件。

原文： Token agent contract implements IBurnRedeemable interface which includes onTokenBurned() method and Redeemed event. This contract also needs to implement ERC-165 standard, specifically it’s supportsInterface() method which shall respond affirmatively to requests about supporting IBurnRedeemable interface. Token controller calls onTokenBurned() method and emits Redeemed event once the burn procedure is complete.

N、 B.：

原文： N.B.:

* 如果代币代理合约不支持IBurnRedeemable或不通过ERC-165进行广告，则“烧毁证明”交易将失败

原文： If the Token agent contract doesn’t support IBurnRedeemable or doesn’t advertise it via ERC-165, the Proof-of-Burn transaction will fail

* 要使刻录交易成功，必须通过ERC-721标准“批准”方法将令牌代理合同批准为特定XENFT令牌ID的操作员，否则交易将失败。

原文： For burn transaction to succeed, Token agent contract must be approved as an operator for as specific XENFT token ID via ERC-721 standard `approve` method, otherwise transaction will fail.