Address Name System: Wallet Ownership Resolution Based on Reliable Self-Declaration

Marcelo Soares *and* Rostand Costa

***Abstract*— Distributed ledger technologies have become popular through the advent of cryptocurrencies, especially Bitcoin, bringing new applicabilities and new challenges. Even though privacy and anonymity are desirable attributes of their users, there is ample evidence that, in some cases, such attributes may be dispensable and digital wallet ownership could be known. In this paper, we introduce a wallet ownership resolution mechanism based on reliable self-declaration of wallet address holders, called Address Name System (ANS). The address-entity mapping public service proposed here relies on special certificates voluntarily populated and verifiable in an independent way. A prototype of this service was implemented as a proof of concept of the proposal and a functional validation of its operations was carried out.**

***Index Terms*— Distributed Ledger Technology, Blockchain, Address-Entity Mapping, Entity Identification, Digital Wallet Ownership Resolution.**

I. INTRODUÇÃO

As tecnologias de livro-razão distribuído, ou DLT (do inglês *Distributed Ledger Technologies,* no singular) estão em constante amadurecimento e crescente utilização, seja no setor privado, ou mesmo por governos e instituições públicas. Projeções apontam para uma expectativa de continuidade no crescimento da utilização de tais tecnologias. Segundo a *International Data Corporation* (IDC), espera-se que o gasto anual com DLTs chegue a 9,7 bilhões de dólares em 2021 [1]. Desde as suas primeiras implementações até os dias atuais, as DLTs são exploradas por uma gama de aplicações e comunidades, onde a privacidade e anonimato são atributos desejáveis por parte dos usuários. Por outro lado, o anonimato padrão das DLTs, para alguns contextos e cenários, pode enfrentar barreiras regulatórias que precisam ser removidas para que uma aplicação baseada em DLT, financeira ou não, pode ser adotada em alguns países [2]. Embora estejam surgindo evidências de que o anonimato pode, em alguns casos, ser dispensável, como também há cenários em que a identificação dos atores envolvidos nas transações se faz necessária, ainda há uma lacuna na literatura acerca de propostas de soluções para a identificação das entidades por trás das chaves públicas associadas às chaves privadas utilizadas para transacionar em DLTs.

Paper submitted 11/09/2020.

Marcelo Soares – Superintendência de Tecnologia da Informação da Universidade Federal da Paraíba, João Pessoa, PB - Brasil (marcelo.soares@sti.ufpb.br).

Rostand Costa – Laboratório de Aplicações de Vídeo Digital da Univerisdade Federal da Paraíba, João Pessoa, PB - Brasil (rostand@lavid.ufpb.br).

Tais chaves públicas são resumidas com funções específicas para gerar um conjunto de caracteres alfanuméricos, conhecidos popularmente como "endereços de carteira", utilizados para representar os usuários dentro do escopo das DLTs. O requisito de identificação dos atores pode não ser um problema para as DLTs privadas, uma vez que apenas atores selecionados podem compor a rede [3].

Entretanto, um dos princípios das DLTs é a confiança provida pela rede aos seus usuários. Em certos casos, decisões de projetos apontam para uma preferência por utilização de redes públicas, uma vez que os registros de tais DLTs são públicos e qualquer interessado pode auditar as transações [4]. Redes públicas com muitos nós são consideradas mais seguras já que estão menos vulneráveis aos ataques de 51%, que ocorre quando um participante detém mais de 50% de poder de processamento da rede, podendo comprometer a integridade dos dados do livro-razão. As redes públicas são abertas para a participação por qualquer interessado, sem mecanismos para a identificação dos usuários que as compõem.

Neste sentido, este trabalho apresenta os resultados de um projeto de pesquisa focado na investigação da utilização de tecnologias consolidadas, a exemplo de certificação e assinatura digital, para o provimento de um modelo que garanta a associação confiável entre endereços de carteiras e entidades do mundo real, contribuindo assim para a evolução e amadurecimento das DLTs, permitindo a sua exploração por uma gama maior de aplicações.

II. TECNOLOGIAS DE LIVRO-RAZÃO DISTRIBUÍDO

Para a completa compreensão da estratégia adotada, faz-se necessário uma breve revisão sobre alguns dos conceitos e tecnologias.

O estudo das tecnologias livro-razão distribuído se inicia com a primeira concepção publicada em 2018 no artigo "Bitcoin: A Peer-to-Peer Electronic Cash System", por uma figura anônima que assinava como Satoshi Nakamoto. Nakamoto levantou a necessidade de um sistema de pagamento baseado em prova criptográfica através de uma rede peer-to-peer ao invés de uma entidade intermediadora [5]. A tecnologia por trás dessa abordagem descrita por Nakamoto foi denominada *blockchain*. Posteriormente, outras tecnologias foram implementadas, surgindo o termo *Distributed Ledger Technology* (DLT) [6]. A atualidade do tema em questão reflete na dificuldade de clareza na terminologia. Muitas vezes os termos *Distributed Ledger Technology* e *Blockchain* se conflitam, como também é possível encontrar diferentes abordagens na implementação de DLTs [3].

De acordo com Bech em [7], um livro-razão distribuído refere-se aos protocolos e infra-estrutura de suporte que permitem que computadores em diferentes locais proponham e validem transações e atualizem registros de maneira sincronizada em uma rede.

O grande alavancamento da popularidade das DLTs está diretamente relacionado à sua utilização pelas criptomoedas, que propõem uma abordagem diferente ao modelo centralizado até então utilizado na nossa sociedade.

Em geral, os termos privacidade e anonimato estão automaticamente incluídos dentro do escopo da utilização de DLTs. Normalmente, o modelo de funcionamento das criptomoedas tenta prover o anonimato dos atores envolvidos, e por consequência, a privacidade em suas ações. Para um cenário onde as transações devem ser públicas, a privacidade pode ser alcançada mantendo anônima a propriedade de uma chave pública. Nesse contexto, o termo pseudônimo digital é frequentemente usado para designar uma chave pública associada a uma chave privada de propriedade desconhecida [8, 9].

A. *Criptografia em DLTs*

Ao detalhar o funcionamento de uma DLT, é possível perceber que diversos elementos da criptografia se fazem presentes. Funções de dispersão criptográfica ou funções *hash* são amplamente utilizadas para a representação de identificadores de transações e blocos. Outro elemento da criptografia presente em DLTs é a árvore de Merkle, que corresponde a um tipo de estrutura de dados que utiliza uma informação resumida para garantir a integridade de um conjunto de dados de uma árvore binária. A técnica consiste basicamente em concatenar valores *hash* de folhas vizinhas recursivamente até que se chegue ao *hash* de uma folha raíz, chamado de *hash* raiz [10].

O processo de autenticação na rede para a escrita no *ledger* se dá através de outro método criptográfico: a assinatura digital. A posse de uma chave privada permite ao participante transacionar em uma DLT ou mesmo participar do processo de consenso distribuído [5]. Na Bitcoin, a forma mais comum de enviar uma transação e escrevê-la na *blockchain* é a *P2PKH* (acrônimo de *Pay To Public Key Hash*). Neste método de criação de transações, um emissor envia em um campo de saída da transação chamado *Pubkey Script* uma instrução que deverá ser atendida pelo receptor, representado por um *hash* de chave pública, para que este possa reivindicar o montante no momento de encadear uma nova transação. No momento em que o receptor deseja transacionar o montante recebido e se tornar um emissor de uma nova transação, o mesmo deverá inserir uma entrada na transação chamada *Signature Script*, que contém uma assinatura digital realizada com a sua chave privada e a respectiva chave pública cujo o seu *hash* é o mesmo inserido no *Pubkey Script*. Concatenando o *Pubkey Script* e o *Signature Script*, cada minerador pode executar o procedimento para a validação da transação antes de tentar inseri-la em um bloco [11].

III. TRABALHOS RELACIONADOS

Em [12], Orman traz uma discussão sobre a dificuldade em gerenciar diferentes identidades digitais, propondo que cada pessoa possua uma identidade digital única, que seja padronizada para ser utilizada por diferentes serviços, com um controle de quais dados poderão ser compartilhados a depender do serviço que necessite da identificação. O autor faz críticas aos modelos baseados em infraestrutura de chave-pública (PKI do inglês *Public Key Infrastructure*) e sugere que tecnologias de livro-razão distribuído, a exemplo de *blockchain*, podem ser utilizadas para a construção de novos modelos de identidades digitais. O autor fortalece seus argumentos referenciando um projeto desenvolvido pelo MIT para a emissão de certificados digitais utilizando *blockchain*.

Lyons et al. consideram que o principal problema com as identidades digitais atualmente, é que elas são, em grande parte, centralizadas [13]. Neste sentido, os autores propõem um modelo de identidade descentralizada, também conhecida como identidade auto-soberana, cujo os usuários criam as suas identidades e adicionam informações a partir de outras fontes, em tese, confiáveis. Os autores mencionam que *blockchain* pode ser uma poderosa solução para diferentes aspectos de identidades decentralizadas.

Um outro projeto desenvolvido no laboratório MIT Media Lab do Instituto de Tecnologia de Massachusetts deu origem a um padrão para a criação de aplicações para a emissão e verificação de registros sobre entidades (neste caso, pessoas físicas), chamado *Blockcerts* [14]. Na *Blockcerts*, reivindicações de afirmações podem ser solicitadas e realizadas entre atores. Basicamente, um emissor faz uma afirmação sobre um destinatário, em um artefato chamado de certificado, seguindo a sintaxe do padrão *Open Badges*. É gerado um *hash* deste certificado que é registrado em uma *blockchain* (Bitcoin ou Ethereum). O certificado então adiciona o *hash* da transação dentro do seu campo signature com um identificador da respectiva blockchain. O *Blockcerts* é *open-source* e pode ser utilizado em projetos de pesquisa ou implementações comerciais. Um ponto interessante nessa solução, é que os atores utilizam endereços de uma DLT como seus identificadores. A *Blockcerts* deixa claro em sua página de FAQ (do inglês *Frequently Asked Question*) que não é capaz de provar a identidade de um indivíduo ou emissor e que não certifica o mapeamento de chaves públicas para indivíduos ou organizações, apontando claramente a problemática tratada neste trabalho.

Durante a leitura de trabalhos relacionados com o tema em questão, percebeu-se que alguns trabalhos apontam a necessidade de identificação de usuários de tecnologias de livro-razão distribuído, sobretudo as instâncias públicas, com reconhecimento legal, isto é, que aplicações baseadas em *blockchain* possam identificar seus usuários valendo-se de modelos consolidados e amparados por legislações vigentes. Os casos abaixo, ambos no Brasil, ilustram essa demanda.

Júnior et al. explícita um cenário onde há a necessidade de associação entre endereços de carteiras e entidades do mundo real [15]. O referido trabalho apresenta uma proposta de criação de uma representação de um ativo digital apelidado de

BNDESToken, em uma infraestrutura de blockchain para rastrear os recursos do Banco Nacional de Desenvolvimento Econômico e Social. Uma das premissas da proposta é que apenas pessoas jurídicas detentoras de um certificado digital e-CNPJ podem receber o BNDESToken. Os autores citam como um pré-requisito para a implementação da proposta, a existência de um serviço que forneça o mapeamento entre endereços de carteiras em uma *blockchain* e pessoas jurídicas do Brasil.

Costa et al. também apresenta um serviço que demanda que as partes envolvidas sejam identificadas de uma forma segura para dar legitimidade as transações [16]. O serviço, chamado de RAP, combina o uso de DLTs, certificação digital e preservação digital para a criação de uma plataforma, escalável e agnóstica, especializada no registro, autenticação e preservação de documentos digitais. Como prova de conceito da plataforma proposta, foi feita a construção de um serviço público para registro e verificação digital da autenticidade de documentos acadêmicos. No protótipo, o registro dos diplomas acadêmicos pode ser feito em duas das DLTs mais populares, Bitcoin e Ethereum, através de uma transação entre a carteira da instituição emissora (uma IES) e a carteira da instituição autenticadora (Ministério da Educação, por exemplo). Neste caso, a publicização inequívoca da propriedade dos endereços de carteiras digitais em pauta poderia garantir a transparência e a segurança das transações de registro.

A comunidade Bitcoin introduziu uma proposta de melhoria do protocolo bitcoin (BIP do inglês *Bitcoin Improvement Proposal*) descrevendo um protocolo de pagamento, que estrutura os dados das transações para evitar erros [17]. O BIP 70 utiliza uma assinatura digital realizada com uma chave associada a um certificado digital X.509 de uma PKI cujo a confiança é baseada em terceiros confiáveis, para identificar o endereço de recebimento no momento da transação. No entanto, como a abordagem foi inicialmente planejada para os comerciantes, percebemos que para a operação do BIP 70 é necessário que cada ator que deseje se identificar na rede Bitcoin faça uma implementação código que forneça um serviço para atender pedidos de *software* de carteira.

Outra abordagem semelhante foi encontrada na carteira *Bitpay* [18]. Essa abordagem amplia o BIP 70 sugerindo o uso de assinaturas com algoritmo de curva elíptica (ECDSA do inglês *Elliptic Curve Digital Signature* Algorithm), o mesmo algoritmo utilizado pelas principais DLTs, como uma alternativa aos certificados digitais X.509. O protocolo especifica rotas para a distribuição das chaves públicas com descrição dos respectivos donos, como também para a distribuição de chaves PGP utilizadas para as assinaturas das chaves públicas utilizadas no protocolo de pagamento.

Além dos exemplos de aplicação citados, inúmeras outras áreas podem se beneficiar do uso de DLTs, sobretudo quando o anonimato pode ser flexibilizado nos moldes propostos pelo ANS. Dentre elas, podemos citar inúmeros casos de uso potenciais na indústria da construção [19], identificação de atores na troca de informações sobre saúde e bem estar de terceiros [20] e como apoio no rastreio de transações em DLTs oriundas de atividades criminosas [21].

IV. ADDRESS NAME SYSTEM

A solução proposta deve contribuir com a dinâmica das DLTs públicas, provendo uma forma segura de identificar atores associados a endereços de carteiras digitais. Por ter o funcionamento semelhante ao do consolidado serviço de tradução de nomes em recursos, Domain Name System (DNS), utilizou-se a nomenclatura Address Name System (ANS) para fazer referência ao sistema proposto neste trabalho.

Um dos requisitos a serem atendidos para a identificação segura, é a confiabilidade da associação entre os endereços e entidades. Isto é, uma associação confiável, segura, e reconhecida como legítima. Muitos governos consideram a PKI como uma tecnologia para a ligação entre entidades e chaves públicas representadas em certificados digitais, com o objetivo de comunicações digitais garantidas, verificáveis e seguras [22]. No Brasil, um documento eletrônico assinado por uma chave emitida por uma Autoridade Certificadora (AC) da cadeia ICP-Brasil possui validade jurídica [23]. A solução proposta utiliza a combinação de assinaturas digitais para prover a identificação: de um lado, uma assinatura digital realizada por uma chave cujo a associação com uma entidade do mundo real é assegurada por uma autoridade confiável, neste caso, uma autoridade certificadora, e de outro lado, uma outra assinatura digital realizada com uma chave privada de uma conta de uma DLT. As assinaturas são depositadas em um artefato, análogo a uma credencial verificável, no entanto, em uma sintaxe bem definida para a assinatura digital em documentos XML, o XMLDSig. A confiança na associação entre a entidade e chave pública do certificado digital é provida por uma infraestrutura de chave pública, uma vez que uma autoridade certificadora realizou os procedimentos necessários para a verificação da identidade.

A. Declaração de Posse de Endereços de Carteiras Digitais

Em uma visão geral, a associação deve ser feita de forma que seja possível comprovar a identidade de uma pessoa/entidade, e comprovar que essa pessoa/entidade é proprietária da chave privada de um determinado endereço. Esta associação é feita através de dois passos: i) **prova de posse** e ii) **prova de identidade**. A **prova de posse** do endereço de carteira indica se a pessoa/entidade possui acesso à chave privada do endereço de carteira para transacionar em uma DLT. A comprovação de posse é feita com a assinatura digital de uma estrutura de dados utilizando a chave privada correspondente à chave pública do endereço reivindicado. A **prova de identidade** tem como objetivo obter a comprovação de que quem está pedindo a associação é quem diz ser e é feita com a assinatura digital realizada com a chave associada a um certificado digital. Uma premissa fundamental do ANS é que tais provas sejam autocontidas e possam ser verificadas de forma autônoma por qualquer interessado em qualquer tempo.

Entretanto, caso utilizadas separadamente, as assinaturas para prova de posse e prova de identidade não fariam qualquer associação entre os proprietários de suas respectivas chaves. É necessária, portanto, a vinculação entre a prova de posse do endereço e a prova de identidade de modo a garantir que o proprietário da chave privada de acesso ao endereço de carteira

é o mesmo proprietário da chave privada que o identifica através do certificado digital. Esta associação é feita com o uso de um documento XML específico, chamado de **ANS Certificate**. A estrutura do **ANS Certificate** (Fig. 1) é baseada na estrutura de um certificado digital X.509, com uma seção de dados a serem assinados, um algoritmo de assinatura e a assinatura digital [24]. Ela foi modelada para fazer referência tanto ao certificado digital do usuário quanto ao endereço da carteira. Para referenciar o certificado digital, o documento contém os campos *Distinguished Names* da AC emissora e o *serial number* do certificado, que o identifica unicamente dentre os certificados emitidos pela AC. Tais atributos são agrupados no elemento <*EntityCertificate*> da seção <*ToBeSigned*>, como pode ser visto na Fig. 1.

```
▼<ANSCertificate>
  ▼<ToBeSigned>
     <DLTInstance>Bitcoin</DLTInstance>
     <Address>1731jmNNp7rTur9UhpGeDK1BkaJVManUsd</Address>
    ▼<EntityCertificate>
       <Type>X.509</Type>
      ▼<Issuer>
         <C>BR</C>
         <O>ICP-Brasil</O>
         <OU>Secretaria da Receita Federal do Brasil - RFB</OU>
         <CN>AC Certisign RFB G5</CN>
       </Issuer>
       <SerialNumber>113229039 ———————— 4891971238</SerialNumber>
     </EntityCertificate>
     <ExpirationDate>2019-08-02 20:15:07</ExpirationDate>
   </ToBeSigned>
  ▶<DLTSignature xmlns="http://www. ———————— /xmldltsig#">...</DLTSignature>
  ▶<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">...</Signature>
 </ANSCertificate>
```

Fig. 1. Exemplo de um ANS Certificate;

Além dos dados de referência ao certificado da pessoa/entidade, a seção <*ToBeSigned*> de um **ANS Certificate** contém ainda os atributos <*DLTInstance*>, <*Address*>, e <*ExpirationDate*>, onde os dois primeiros especificam a instância da DLT e o endereço da carteira, respectivamente, e o último indica a data de validade do **ANS Certificate**. Há ainda o elemento <*DLTSignature*> usado para armazenar a assinatura digital usada como prova de posse e, finalmente, o elemento <*Signature*>, usado para armazenar a assinatura digital usada em nosso contexto como prova de identidade. As assinaturas são acrescentadas ao final do documento conforme a estratégia de assinatura *Enveloped*. Na estratégia de assinatura *Enveloped*, o conteúdo da assinatura é o próprio documento XML e o valor da assinatura é inserido ao final do documento juntamente com o certificado digital do assinante, gerando um artefato final que contém todos os elementos necessários para a verificação de autenticidade da informação criptografada [25]. O elemento <*Signature*> utiliza a estrutura em conformidade com o consolidado padrão internacional W3C, que armazena além da assinatura digital, o certificado associado à chave privada utilizada [25]. O elemento <*DLTSignature*> contém o algoritmo usado para a assinatura, a chave pública referente ao endereço de carteira e a assinatura digital, conforme modelado no *XML Schema Definition* (XSD) e apresentado na Fig. 2.

```
▼<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.0.0">
 ▼<xs:element name="DLTSignature">
  ▼<xs:complexType>
   ▼<xs:sequence>
      <xs:element name="Algorithm" type="xs:string"/>
      <xs:element name="PublicKey" type="xs:string"/>
      <xs:element name="SignatureValue" type="xs:string"/>
    </xs:sequence>
   </xs:complexType>
  </xs:element>
 </xs:schema>
```

Fig. 2. *XML Schema Definition* do elemento DLTSignature;

Para a produção de um ANS Certificate, a pessoa/entidade inicialmente assina digitalmente a seção <*ToBeSigned*> com a chave privada do endereço e, em seguida, tanto a seção <*ToBeSigned*> quanto a seção <*DLTSignature*> são assinadas com a chave privada do certificado, gerando um pacote que contém o ANS Certificate duplamente assinado, a chave pública associada ao endereço para a verificação da prova de posse e o certificado digital com a respectiva chave pública para a verificação da prova de identidade. Como a assinatura para a prova de identidade é feita com a chave privada do certificado digital que é referenciado no próprio documento, é formado um elo entre a prova de posse do endereço de carteira e a prova de identidade. Este fluxo está ilustrado na Fig. 3.

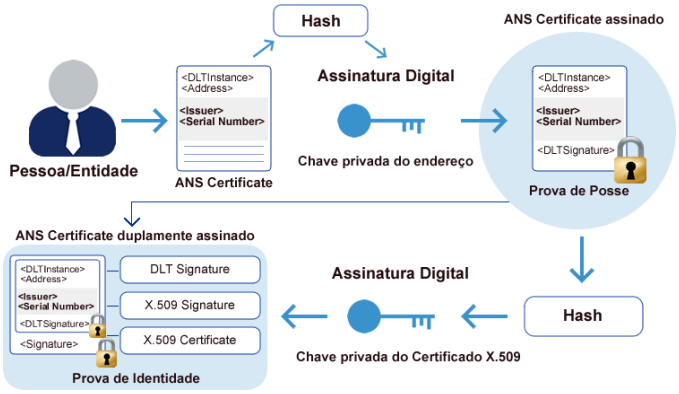

Fig. 3. Fluxo de assinatura de um ANS Certificate;

B. *Verificação de Posse de Endereços de Carteiras Digitais*

A validação das assinaturas e, consequentemente, da associação entre pessoa/entidade e endereço de carteira, é feita em um processo inverso que usa as respectivas chaves públicas do certificado digital e do endereço de carteira. Utilizando a chave pública contida no certificado digital utilizado, é possível conferir a autenticidade da prova de identidade, e associar o ANS Certificate ao proprietário do certificado. Da mesma forma, é possível verificar a assinatura digital da prova de posse utilizando a chave pública do endereço de carteira e associar o ANS Certificate ao proprietário do endereço. A Fig. 4 ilustra o fluxo para a verificação das provas necessárias para a validação da associação endereço-entidade. É possível observar que representações resumidas (*hash*) dos artefatos assinados são geradas a partir das respectivas chaves públicas para serem confrontadas e validadas, uma vez que as assinaturas também foram feitas sobre valores de *hash*.

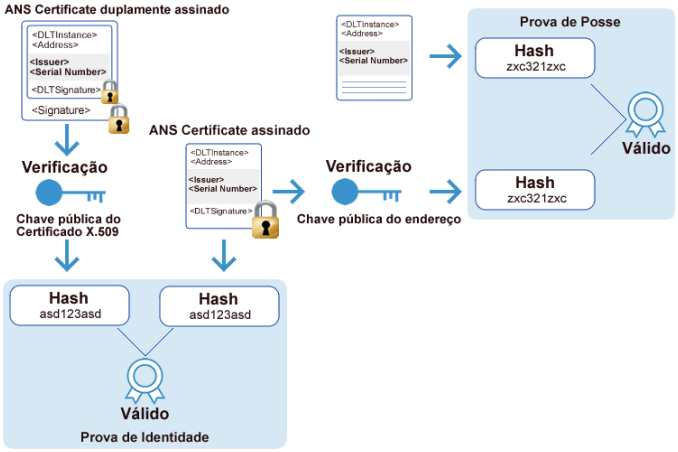

Fig. 4. Fluxo de verificação de um ANS Certificate;

V. METODOLOGIA

A metodologia adotada neste trabalho é a metodologia de Construção (*Build*). Nesta metodologia, almeja-se comprovar uma hipótese através da sua construção. Um protótipo functional do ANS foi desenvolvido e concebido com uma arquitetura formada pelos seguintes componentes: *ANS Client, ANS Server e ANS Repository*. O ANS Client consiste em uma aplicação cliente que recebe dados de entrada, como dados para acesso às chaves privadas do certificado digital e do endereço de carteira e realiza as devidas assinaturas para a geração de um ANS Certificate. O ANS Client também realiza o registro de ANS Certificates produzidos em uma implementação *server-side* denomidada ANS Server. A Fig. 5 mostra a interface do ANS Client no caso de uso de geração e registro de ANS Certificate. O ANS Server por sua vez conduz o processo de registro e disponibiliza uma API *RESTful* para a consulta de associação entidade-endereço. Para o registro no ANS Server, basicamente são realizadas as seguintes tarefas: validação das assinaturas com as respectivas chaves públicas, verificação do elo entre as duas provas, inserção de um registro em sua tabela de mapeamento e armazenamento do ANS Certificate em um repositório distribuído denominado ANS Repository.

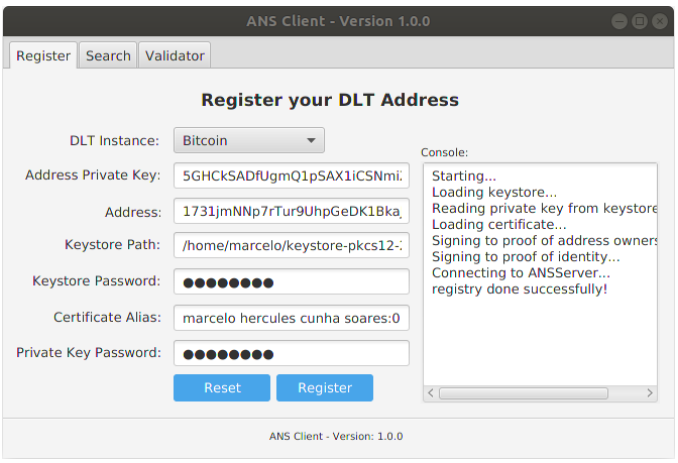

Fig. 5. ANS Client - Geração e registro de ANS Certificate;

O serviço para consulta de associação entidade-endereço recebe como entrada os parâmetros: *dlt-instance* e *address*, correspondentes à instância da DLT e o endereço que deseja ser pesquisado respectivamente. Ao receber uma solicitação de consulta, o ANS Server pesquisa na sua tabela de mapeamento a partir dos parâmetros informados. Caso haja um registro, é realizada a obtenção do respectivo ANS Certificate no ANS Repository, a partir do *path* armazenado na base do ANS Server, com a validação das assinaturas novamente antes de devolver o ANS Certificate para o solicitante.

A. Experimentos

A realização dos experimentos foi feita em dois cenários. Primeiramente foram feitos experimentos em um ambiente local, com utilização do ANS Client para a geração do ANS Certificate, uma instância de um ANS Server executada localmente uma DLT Ethereum inicializada localmente sem conexão com outros nós. Em um segundo caso, foram feitos experimentos em instâncias públicas reais das DLTs Bitcoin e Ethereum. Para os experimentos, foi utilizado um certificado digital real emitido por uma Autoridade Certificadora da ICPBrasil com o valor de *Subject Name*: MARCELO HERCULES CUNHA SOARES:071XXXXXXX1. Os caracteres "X" ocultam um número de CPF real.

1) Integração com Block Explorer

Para facilitar a visualização da identidade dos usuários envolvidos no contexto de uma determinada aplicação que utiliza de uma DLT, foi feita a integração do ANS com uma ferramenta de *Block Explorer*. Um *Block Explorer* é uma ferramenta que fornece informações sobre blocos, transações e endereços de DLTs, desde uma visão mais macro (últimos blocos minerados), até uma visão mais detalhada (transações de um determinado endereço).

Para a integração com o ANS em uma plataforma Ethereum, foi utilizado o *Block Explorer* de código aberto *Ethnamed Block Explorer*, conectado a *blockchain* local da Ethereum através das interfaces de conexão RPC da biblioteca javascript *web3*. O código da ferramenta foi customizado para fazer chamadas ao serviço de consulta do ANS. Na tela inicial do *Block Explorer* é possível visualizar os últimos blocos minerados, isto é, adicionados à cadeia. Dentro de cada bloco é possível visualizar os dados das transações contidas. A maneira tradicional de visualização de dados de transações em tais ferramentas é a exibição do endereço de carteira/contrato de origem, o valor do ativo transacionado, e o endereço de carteira/contrato de destino. Para cada transação contida dentro de um bloco, o *Block Explorer* faz uma chamada à API *RESTful* do ANS Server, passando os parâmetros necessários para o serviço de consulta, neste caso, o identificador da rede Ethereum e um endereço envolvido na transação visualizada. Caso haja uma ocorrência de registro do endereço consultado no ANS Server, o mesmo retorna uma resposta em formato *json* para o *Block Explorer*, com os dados do detentor do endereço, incluindo o CID para acessar o respectivo ANS Certificate.

O *Block Explorer* por sua vez exibe ao lado do endereço, caso haja um retorno do ANS Server, o atributo *Subject Name* do certificado digital contido no ANS Certificate e utilizado para a prova de identidade. Como é possível visualizar na Fig. 6, o

usuário tem a possibilidade de fazer o *download* do ANS Certificate, caso deseje escrever o seu próprio mecanismo de validação, ou mesmo fazer o *download* do Certificado Digital X.509 contido no ANS Certificate para verificar as informações do detentor do endereço, como também informações sobre a Autoridade Certificadora responsável por assinar o certificado. Uma observação importante é que, neste caso, apenas uma das contas possuía uma declaração de posse registrada, isto é, enquanto um participante da transação permite a sua identificação, o outro participante permaneceu de forma anônima, permitindo uma convivência pacífica com os atores que desejem o anonimato.

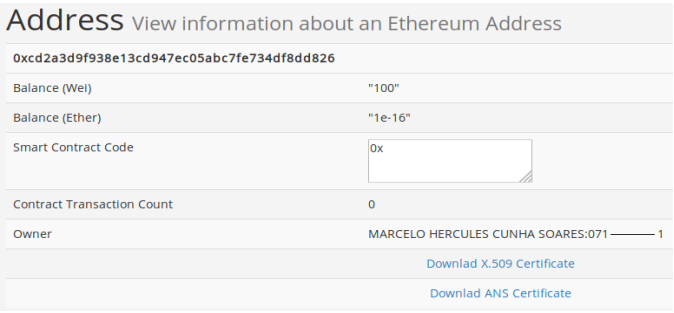

Fig. 6. Visualização de dados do endereço de carteira no *Block Explorer;*

2) *Integração com RAP/SIGAA*

A aplicação de registro, autenticação e preservação de documentos digitais, denominada RAP, desenvolvida no Laboratório de Aplicações de Vídeo Digital (LAVID) da Universidade Federal da Paraíba (UFPB) foi utilizada pela Superintendência de Tecnologia da Informação (STI) da UFPB em um projeto piloto para a emissão de diplomas digitais para alunos concluintes dos cursos de Ciências da Computação e Engenharia da Computação. O serviço RAP, ao receber uma solicitação de registro de documento, além de registrar o documento em sua respectiva tecnologia de preservação, também realiza o registro de uma representação resumida em uma tecnologia de livro-razão distribuído, neste caso, a DLT pública da plataforma Ethereum. Com o registro na *blockchain*, é possível garantir que o documento existia naquele determinado momento em que foi registrado, como também, a sua integridade pode ser verificada por qualquer parte que possua o documento original.

Por sua natureza, espera-se que as informações apresentadas aos usuários no RAP sejam confiáveis. A identificação do ator remetente da transação de registro de diplomas se faz necessária para que as informações fornecidas pelo serviço sejam comprovadamente verdadeiras e verificáveis. No ponto de vista de arquitetura de sistemas, neste caso, o cliente do serviço RAP é o Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA) da UFPB. No portal externo do SIGAA há um link para a autenticação de diplomas digitais, com um formulário para que o usuário realize o *upload* do documento a ser verificado. No momento da verificação, o SIGAA faz uma chamada ao serviço RAP através de uma API *RESTful*, que por sua vez retorna os dados relativos ao registro do diploma no serviço, como também os dados relativos ao registro na DLT, como o *hash* da transação realizada na *blockchain*. Após o retorno dos dados de registro pelo RAP, o SIGAA realiza uma chamada ao ANS Server, passando como parâmetros o identificador da DLT Ethereum e o endereço de carteira utilizado na transação. O ANS Server por sua vez, retorna os dados registrados ao SIGAA, uma vez que uma associação foi encontrada, e por sua vez o SIGAA os exibe em tela para o usuário, conforme pode ser visto na Fig. 7.

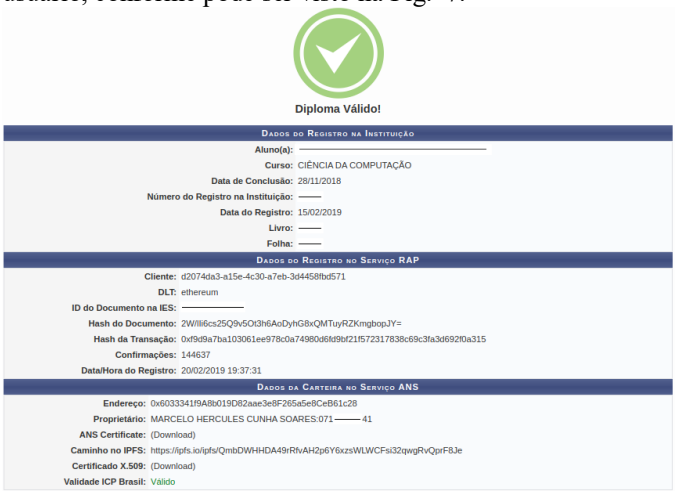

Fig. 7. Interface para validação de diplomas digitais no SIGAA;

3) *Integração on chain*

O ANS também foi projeto para attender a uma categoria de aplicações que estão dentro do escopo das DLTs, denomiada aplicações descentralizadas (dApp). Para essas aplicações, é necessária a utilização de um novo componente denominado ANS Oracle, a fim de popular os dados *on chain* com dados oriundos do escopo externo à rede, providos pelo ANS Server. Este experimento foi dividido em duas etapas. Na primeira etapa foi realizada a integração com um contrato inteligente simples, cujo o papel é representar uma dApp cliente do ANS Oracle. Uma *blockchain* da Ethereum foi inicializada localmente para este experimento e interações com o ANS Oracle foram feitas a partir da IDE *remix*. O *remix* permite estabelecer uma conexão com uma rede Ethereum, seja uma rede local ou remota, como também, fornece uma interface amigável provendo uma interface com os *inputs* necessários para interagir com contratos inteligentes. A Fig. 8 mostra o retorno de uma consulta à função *getEntityByAddress* ao ANS Oracle realizada na interface do Remix com o seu respectivo retorno.

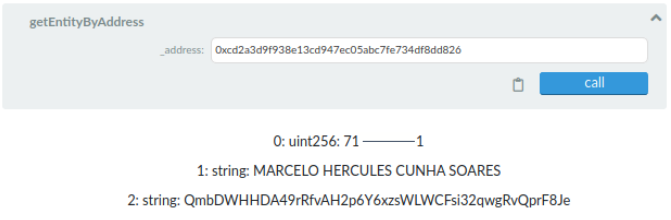

Fig. 8. Retorno de consulta no ANS Oracle na IDE Remix;

Para iniciar o experimento, foi necessário realizar o deploy do ANS Oracle em uma rede Ethereum. Um código utilitário foi escrito em javascript com *nodejs* para o deploy do contrato utilizando a biblioteca *solc.js*, que fornece uma API para as operações do compilador solidity. A estratégia adotada para a

alimentação do ANS Oracle é através da emissão de um evento na dApp que é escutado pelo ANS Listener, através de uma conexão *websocket* entre a aplicação e o nó que executa o *geth*, neste caso, a própria máquina local. O ANS Listener entra em ação quando o evento *EntityRequest* é disparado. O *listener* por sua vez consulta o ANS Server passando como parâmetro o identificador da rede Ethereum e o endereço enviado no evento. Após a resposta do ANS Server, caso uma ocorrência seja encontrada, o *listener* recupera e faz a validação do ANS Certificate no verificador de conformidade de assinatura digital do órgão responsável por manter a infraestrutura de chaves públicas brasileira, o ITI. Em seguida, executa uma transação para o ANS Oracle passando como parâmetros necessários os dados retornados do ANS Server. A utilização do ANS Oracle por dApps clientes se dá através da importação do ANS Oracle nos contratos inteligentes. Desse modo, é possível realizar chamadas a funções do ANS Oracle fazendo invocações pelos respectivos nomes na variável que contém a referência.

O BNDES disponibiliza em seu repositório no GitHub, o código-fonte dos contratos inteligentes pertencentes ao projeto BNDESToken. Para avaliar mais profundamente a viabilidade do ANS Oracle, foi realizado também um experimento de integração com o BNDESToken. Pela falta de documentação no repositório público do projeto, o entendimento do código-fonte foi baseado na interpretação de nomes de variáveis e funções, na leitura de comentários e documentação a nível de código, como também da análise da relação entre os contratos existentes. Os seguintes contratos foram obtidos e implantados: *BNDESRegistry* e *BNDESToken*. Com uma análise do código, percebeu-se que o contrato BNDESRegistry atua como um contrato auxiliar, provendo funcionalidades e dados ao contrato BNDESToken. Associações entre CNPJs e endereços são armazenados em uma variável do tipo *mapping*, que chaveia de endereços (tipo *address*) para uma *struct* definida e chamada *LegalEntityInfo*. A alimentação dessas associações é feita na função *registryLegalEntity*. O contrato BNDESRegistry foi alterado para receber uma referência do ANS Oracle em seu construtor, como também o método *registryLegalEntity* foi alterado para realizar uma consulta a uma instância do ANS Oracle. A Fig. *9* representa o fluxo de alimentação do ANS Oracle.

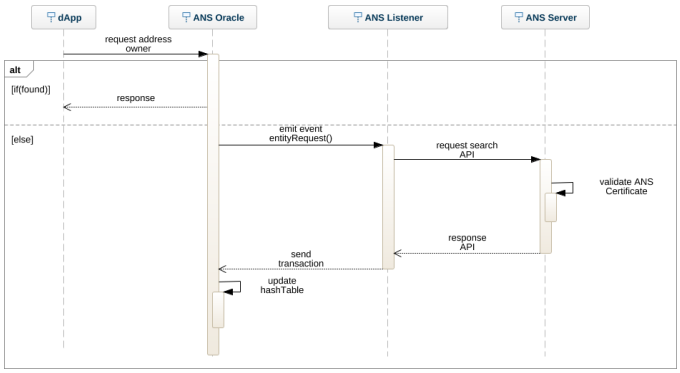

Fig. 9. Fluxo de alimentação de um ANS Oracle;

Uma vez que o ANS Oracle está populado com associações cujo a identidade está representada em um certificado digital validado pela ICP Brasil, o BNDESRegistry só conseguirá popular a sua tabela de associações endereços retornados pelo ANS Oracle, e o BNDESToken por consultar esta associação, só será transferido entre contas cujo a identidade dos proprietários são representadas em certificados do tipo e-CNPJ, conforme requisito apontado no projeto.

VI. CONSIDERAÇÕES FINAIS

Durante o desenvolvimento deste trabalho foi possível observar a dinâmica na evolução das tecnologias de livro-razão distribuído e entender que a sua capacidade de adaptação em diferentes contextos traz também novos desafios. Percebeu-se que o anonimato e a privacidade em determinadas situações podem ser dispensados, abrindo uma grande lacuna para a investigação de soluções que possam ajudar a identificar, legítima e inequivocamente, as entidades por trás de endereços de carteiras digitais.

A solução proposta sugere a utilização de infraestruturas de chave pública existentes para o provimento da confiança no quesito da identificação dos atores por trás dos endereços. Percebeu-se que, com a ampla utilização de PKIs como soluções para identidade com reconhecimento legal, esta tecnologia poderia ser aproveitada para atuar em parte do problema. O modelo proposto para atingir o objetivo de prover uma associação confiável é baseado na utilização de assinatura digital para a geração de um artefato análogo a uma credencial verificável. O mecanismo proposto pode ser utilizado por aplicações onde é necessária a identificação dos atores envolvidos em transações de forma não exclusiva e com suporte a múltiplas instâncias de DLTs. O protótipo desenvolvido como prova de conceito ajudou a demonstrar a viabilidade da declaração espontânea de uma relação endereço-entidade como também a sua recuperação e verificação de forma independente.

Com o intuito de avaliar a proposta, foram realizados experimentos de integração do protótipo funcional com aplicações reais, demonstrando ser uma solução viável, uma vez que é de fácil integração e atinge a finalidade para o qual foi proposto. O ANS pode ser considerado em parte descentralizado, uma vez que partes da solução podem ser consideradas centralizadas, como a utilização de PKI. Diferente de outras plataformas, uma única instância do ANS pode atender a vários usuários, sem a necessidade da instanciação de serviços para cada usuário. O ANS foi concebido para que também possa ser utilizado por aplicações decentralizadas (on-chain), recurso não encontrado em outras soluções.

Em comparação com as abordagens relacionadas existentes, percebe-se algumas vantagens em relação à utilização do ANS. Basicamente, as abordagens citadas na seção III fornecem apenas um protocolo semelhante ao que faz o HTTPS para a internet, sendo diferente da abordagem da ANS que extrapola o contexto da sessão de operação. Por exemplo, o ANS permite a integração com ferramentas de exploração de bloco, permitindo a visualização de todos os envolvidos em transações de um determinado aplicativo baseado em DLT que exige transparência.

As abordagens elencadas na seção III permitem a identificação de destinatários de transações, mas não identificam endereços de remetentes, como a ANS. É

importante mencionar que o caráter voluntário e autoverificável da ANS permite a coexistência pacífica de publicidade e privacidade das carteiras no mesmo DLT.

REFERÊNCIAS

[1] IDC. 2018. New IDC Spending Guide Sees Worldwide Blockchain Spending Growing to $9.7 Billion in 2021. https://www.businesswire.com/news/home/20180124005949/en/New-IDC-Spending-Guide-Sees-Worldwide-Blockchain [Online; accessed 04-August-2018].

[2] Priem, R. (Feb, 2020). Distributed ledger technology for securities clearing and settlement: benefits, risks, and regulatory implications. Financ Innov 6, 11. [Online]. Avaliable: https://doi.org/10.1186/s40854-019-0169-6

[3] Advait Deshpande, Katherine Stewart, Louise Lepetit, and Salil Gunashekar. 2017. Distributed Ledger Technologies/Blockchain: Challenges, opportunities and the prospects for standards. Technical Report. British Standards Institution (BSI).

[4] BNDES. 2018. Public Call Notice AARH 05/2018 – BNDES. www.bndes.gov.br/consultablockchain [Online; accessed 04-August-2018].

[5] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. (2008).

[6] Harish Natarajan, Solvej Krause, and Helen Gradstein. 2017. Distributed Ledger Technology (DLT) and Blockchain. (2017). FinTech note; no. 1. Washington, D.C.: World Bank Group. http://documents.worldbank.org/curated/en/177911513714062215/DistributedLedger-Technology-DLT-and-blockchain. [Online; accessed 06-August-2018].

[7] Morten Linnemann Bech and Rodney Garratt. 2017. Central bank cryptocurrencies. https://www.bis.org/publ/qtrpdf/r_qt1709f.htm

[8] Simon Taylor, Richard G Brown, Vili Lehdonvirta, Robleh Ali, Angela Sasse, Phil Godsiff, Phil Godsiff, Catherine Mulligan, and Patrick Curry. 2016. Distributed Ledger Technology: beyond block chain. Technical Report. Government Office for Science.

[9] Arvind Narayanan and Jeremy Clark. 2017. Bitcoin's Academic Pedigree. Commun. ACM 60, 12 (Nov. 2017), 36–45. https://doi.org/10.1145/3132259

[10] Ralph C. Merkle. 1982. Method of providing digital signatures. https: //patentimages.storage.googleapis.com/69/ab/d9/2ff9f94fada6ea/US4309569.pdf [Online; accessed 04-February-2020].

[11] Bitcoin. 2018. Transactions Guide - Bitcoin. https://bitcoin.org/en/you-need-to-know https://bitcoin.org/en/transactions-guide. [Online; accessed 04-August-2018].

[12] H. Orman. 2018. Blockchain: The Emperors New PKI? IEEE Internet Computing 22, 2 (Mar 2018), 23–28. https://doi.org/10.1109/MIC.2018.022021659

[13] Tom Lyons, Ludovic Courcelas, and Ken Timsit. 2019. Blockchain and Digital Identity. Technical Report. European Union Blockchain Observatory Forum. https://www.eublockchainforum.eu/reports. [Online; accessed 12-February2019].

[14] Ralph C. Merkle. 1982. Method of providing digital signatures. https://patentimages.storage.googleapis.com/69/ab/d9/2ff9f94fada6ea/US4309569.pdf [Online; accessed 04-February-2020].

[15] Gladstone Moisés Arantes Júnior, José Nogueira D'Almeida Jr., Marcio Teruo Onodera, Suzana Mesquita de Borba Maranhão Moreno, and Vanessa da Rocha Santos Almeida. 2018. BNDESToken: Uma Proposta para Rastrear o Caminho de Recursos do BNDES. Workshop em Blockchain: Teoria, Tecnologias e Aplicações (WBlockchain_SBRC) 1, 1/2018 (2018). https://portaldeconteudo.sbc.org.br/index.php/wblockchain/article/view/2355

[16] Rostand Costa, Daniel Faustino, Guido Lemos, Ademir Queiroga, Cláudio Djohnnatha, Felipe Alves, Jordan Lira, and Mateus Pires. 2018. Uso Não Financeiro de Blockchain: Um Estudo de Caso Sobre o Registro, Autenticação e Preservação de Documentos Digitais Acadêmicos. Workshop em Blockchain: Teoria, Tecnologias e Aplicações (WBlockchain-SBRC) 1, 1/2018 (2018). http: //ojs.sbc.org.br/index.php/wblockchain/article/view/2356

[17] Gavin Andresen and Mike Hearn. 2013. Bitcoin Improvement Proposal 70. https: //github.com/bitcoin/bips/blob/master/bip-0070.mediawiki [Online; accessed 20-August-2019].

[18] Bitpay. 2019. JSON Payment Protocol Specification V2. https://bitpay.com/docs/ payment-protocol [Online; accessed 21-August-2019].

[19] J. J. Hunhevicz, D. M. Hall, (Aug, 2020). Do you need a blockchain in construction? Use case categories and decision framework for DLT design options, Advanced Engineering Informatics, Volume 45, 2020, 101094, ISSN 1474-0346, https://doi.org/10.1016/j.aei.2020.101094

[20] J. Kleinschmidt, P. S. R. Garcia. 2020. Sharing Health and Wellness Data with Blockchain and Smart Contracts. (May, 2020). Vol. 18 No. 6 (2020): Ordinary Issue. https://latamt.ieeer9.org/index.php/transactions/article/view/1173

[21] V. G. R. Macedo. 2020. WannaCry on Bitcoin Blockchain: A Tracking Study Case. (Dec, 2019). Vol. 17 No. 7 (2019): Ordinary Issue. https://latamt.ieeer9.org/index.php/transactions/article/view/153

[22] Ali M. Al-Khouri. 2012. PKI in Government Digital Identity Management Systems. 4-21 pages. https://www.ica.gov.ae/userfiles/ePractice%20Journal_Volume_14_FINAL_28.2.2012_Part3.pdf.

[23] Brasil. 2001. Medida Provisória No 2.200-2, de 24 de Agosto de 2001. http: //www.planalto.gov.br/ccivil_03/mpv/antigas_2001/2200-2.htm [Online; accessed 10-August-2018].

[24] Sharon Boeyen, Stefan Santesson, Tim Polk, Russ Housley, Stephen Farrell, and Dave Cooper. 2008. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. RFC 5280. https://doi.org/10.17487/ RFC5280

[25] Mark Bartel, John Boyer, Barb Fox, Brian LaMacchia, and Ed Simon. 2015. XML Signature Syntax and Processing. https://www.w3.org/TR/xmldsig-core2/ [Online; accessed 05-August-2018].

Marcelo H. C. Soares possui os títulos de Tecnólogo em Sistemas Para Internet (2011) pela Faculdade de Tecnologia da Paraíba, Especialista em Engenharia de Sistemas (2015) e Mestre em Informática (2020) pela Universidade Federal da Paraíba onde também compõe o corpo técnico da instituição, atuando na Gerência de Bases de Dados da Superintendência de Tecnologia da Informação (STI). Tem experiência na área de Ciência da Computação, com ênfase em Sistemas de Computação. Possui ampla experiência em engenharia de software. Atuou em projetos de pesquisa na exploração de tecnologias de livro-razão distribuído fora do contexto financeiro.

Rostand E. O. Costa recebeu os títulos de Bacharel em Ciência da Computação pela Universidade Federal da Paraíba - UFPB (1988), Especialista em Engenharia Elétrica (1989) e Mestre em Informática (1999) pela mesma Universidade e Doutor em Ciência da Computação pela Universidade Federal de Campina Grande - UFCG, em 2013. Atualmente é pesquisador no Laboratório de Aplicações de Digital Video (LAVID) da UFPB e pesquisador associado no Laboratório de Sistemas Distribuídos (LSD) da UFCG. Desenvolve pesquisas na área de Sistemas Distribuídos e Engenharia de Software em geral, com interesse especial em computação na nuvem, blockchain, computação de alta vazão e aplicações de vídeo, cinema e TV digital. Também atua, há mais de 25 anos, como consultor em engenharia de software e aplicativos computacionais, com experiência em orgãos governamentais, bancos e empresas do segmento de educação e saúde suplementar.