

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Pedro Paulo Vilela Silva

**Estudo Experimental Comparativo entre
Content Centric Networking e Entity Title
Architecture**

Uberlândia, Brasil

2017

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Pedro Paulo Vilela Silva

**Estudo Experimental Comparativo entre Content Centric
Networking e Entity Title Architecture**

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido parcial à obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Flávio de Oliveira Silva, Ph.D

Universidade Federal de Uberlândia – UFU

Faculdade de Ciência da Computação

Bacharelado em Ciência da Computação

Uberlândia, Brasil

2017

Pedro Paulo Vilela Silva

Estudo Experimental Comparativo entre Content Centric Networking e Entity Title Architecture

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido parcial à obtenção do grau de Bacharel em Ciência da Computação.

Trabalho aprovado. Uberlândia, Brasil, 3 de agosto de 2017:

Prof. Flávio de Oliveira Silva, Ph.D
Orientador

Prof. Pedro Frosi Rosa, Ph.D
Convidado 1

**Prof. Ronaldo Castro de Oliveira,
Ph.D**
Convidado 2

Uberlândia, Brasil
2017

Dedico esse trabalho à minha mãe que sempre me incentivou e apoiou a minha formação

Agradecimentos

Agradeço a minha família pelo suporte e amor.

Agradeço a minha namorada pelo amor, companheirismo e dedicação.

Agradeço ao Flávio Silva pela orientação e apoio para a conclusão desse trabalho e para a minha formação.

Agradeço aos meus amigos pela força e apoio.

Resumo

A pesquisa em redes de Internet do Futuro é fundamental para identificar e validar uma nova rede, mais eficiente, eficaz e capaz de atender os novos conjuntos de requisitos da rede. Nesse estudo será feita uma comparação entre duas propostas *clean slate* de arquiteturas de rede do futuro. A primeira abordagem, ETArch, é uma rede centrada no *workflow* capaz de gerenciar e controlar o funcionamento da rede, essencialmente implementando o conceito das redes *Software Defined Networking* (SDN). A segunda arquitetura, CCN, é uma rede centrada no conteúdo cujo o foco é o armazenamento e a distribuição de conteúdos através de nomes. As arquiteturas ETArch e CCN foram implementadas em uma máquina de uso pessoal com o intuito de analisar o comportamento dessas redes sobre uma aplicação de *Chat*. A avaliação das abordagens ETArch e CCN foi feita a partir de um ambiente de teste analisando um conjuntos de métricas.

Palavras-chave: Internet do Futuro, *Entity Title Architecture* (ETArch), *Content-Centric Networking* (CCN), *clean slate*, *Information-Centric Networking* (ICN) .

Lista de ilustrações

Figura 1 – Exemplo da arquitetura <i>OpenFlow</i> (OPENFLOW, 2010)	20
Figura 2 – Exemplo da arquitetura SDN (SDXCENTRAL, 2015)	21
Figura 3 – Exemplo de um nó CCN(a direita) e um nó da Internet atual (a esquerda) (KANG BYUNGRYEOL SIM; LEE, 2012)	22
Figura 4 – Exemplo de nome de um conteúdo na rede CCN	23
Figura 5 – Exemplo dos dois tipos de pacote da rede CCN: <i>Interest</i> e <i>Data</i> (CABRAL; ROTHENBERG, 2013)	23
Figura 6 – Exemplo das estruturas de um nó CCN (AMBIEL; MAGALHÃES, 2013)	24
Figura 7 – Exemplo dos Principais Componentes e Protocolos da rede ETArch (SILVA, 2013)	26
Figura 8 – Exemplo dos Pilha de Protocolo da rede ETArch (CASTILLO et al., 2014)	26
Figura 9 – Exemplo de iniciação de um <i>Workflow</i> na rede ETArch (CASTILLO et al., 2014)	27
Figura 10 – Exemplo da topologia <i>full mesh</i> usada para os testes	30
Figura 11 – Exemplo do fluxo de mensagem que será usado para os testes	32
Figura 12 – Exemplo do <i>Chat CCN</i>	33
Figura 13 – Exemplo da criação da topologia de rede ETArch e o fluxo de caminhos definidos no <i>controller</i>	33
Figura 14 – Exemplo do <i>Chat ETArch</i>	34
Figura 15 – Tamanho dos pacotes do experimento ETArch	34
Figura 16 – Tamanho dos pacotes do experimento CCN	35
Figura 17 – Gráfico gerado para análise de <i>throughput</i> entre ETArch e CCN	35

Lista de abreviaturas e siglas

API	Application Programming Interface
CBE	Container-Based Emulation
CCN	Content Centric Networking
CIDR	Classless Inter-Domain Routing
CLI	Command Line Interface
CS	Content Store
DOS	Denial of Service
DTS	Domain Title Service
DTSA	Domain Title Service Agent
ETArch	Entity Title Architecture
FIB	Forward Information Base
FIBRE	Future Internet Experimentation Between Brazil and Europe
FIND	Future Internet Design
FIRE	Future Internet Research and Experimentation
ForCES	Forwarding and Control Element Separation
GENI	Global Environment for Network Innovations
ICN	Information-Centric Networking
IoT	Internet of Things
IP	Internet Protocol
IPV4	Internet Protocol version 4
IPV6	Internet Protocol version 6
MF	Mobility First
MIH	Media Independent Handover

NAT	Network Address Translation
NDN	Named Data Networking
OFELIA	OpenFlow in Europe Linking Infrastructure and Applications
ONOS	Open Network Operating System
OSI	Open Systems Interconnect
PARC	Palo Alto Research Center
PIT	Pending Interest Table
PoA	Point of Attachment
QoE	Quality of Experience
QoS	Quality of Service
SDN	Software Defined Networking
TCP	Transmission Control Protocol
VoD	Video on Demand
VoIP	Voice over Internet Protocol

Sumário

1	INTRODUÇÃO	10
1.1	Objetivos	12
1.2	Justificativa	12
1.3	Estado da Arte	13
2	MÉTODO	15
3	REFERENCIAL TEÓRICO	17
3.1	Internet do Futuro	17
3.2	<i>OpenFlow e Software Defined Networking (SDN)</i>	19
3.3	<i>Content Centric Networking (CCN) / Named Data Networking (NDN)</i>	22
3.4	ETArch	24
3.5	Trabalhos Correlatos	27
4	AVALIAÇÃO EXPERIMENTAL	29
4.1	Metodologia	29
4.2	Implementação: CCN	30
4.3	Implementação: ETArch	31
4.4	Experimentos e Resultados	31
4.4.1	Experimento com a CCN	32
4.4.2	Experimento com a ETArch	32
4.4.3	Resultados e Discussão	33
5	CONCLUSÃO	37
	Referências	38

1 Introdução

A Internet foi projetada para um cenário diferente dos dias atuais. Isso se deve ao fato da sua constante evolução, motivada pelo crescimento de usuários e a introdução de novos dispositivos, aplicativos e serviços (HANDLEY, 2006). A evolução da Internet trouxe um novo conjunto de exigências e com isso evidenciando a limitação da rede atual em alguns aspectos como endereçamento, mobilidade, segurança, confiabilidade da rede, disponibilidade de serviço, diagnóstico de problemas, gerenciamento da rede, QoS e escalabilidade (MOREIRA et al., 2009; ZAHARIADIS et al., 2011).

A arquitetura da Internet atual não oferece suporte adequado aos requisitos da Internet do Futuro. Nesse contexto, duas propostas são consideradas pela comunidade de pesquisa de rede. A primeira é chamada *clean slate*, ou arquitetura disruptiva que tem o propósito de desenvolver uma nova arquitetura de rede para substituir a atual. A segunda abordagem consiste em evoluir a arquitetura TCP/IP (também conhecido como pilha de protocolos TCP/IP) já implantada. Os protocolos TCP (*Transmission Control Protocol*) (POSTEL, 1981) e IP (*Internet Protocol*) (POSTEL, 1980) é um conjunto de protocolos de comunicação entre computadores em rede (REXFORD; DOVROLIS, 2010).

Como resultado desse fato algumas abordagens *clean slate* estão sendo estudadas para tratar os novos conjuntos de requisitos da Internet do Futuro. Segundo (STERBENZ et al., 2011) os principais desafios para o sucesso das novas arquiteturas são: robustez e disponibilidade, suporte a mobilidade dos nós economicamente viável e rentável, gerenciabilidade com segurança e ser evolutiva.

Com esse problema em mente, uma solução que a comunidade de pesquisa encontrou foi criar um ambiente para facilitar a pesquisa experimental em Internet do Futuro. Nesse contexto, surgiram programas como *Global Environment for Network Innovations* (GENI), *Future Internet Design* (FIND) e *Future Internet Research and Experimentation* (FIRE) com a iniciativa de propor um ambiente de pesquisa com infraestrutura para habilitar, controlar e testar novas arquiteturas de rede do futuro (ELLIOTT, 2008; FARIAS et al., 2011; STUCKMANN; ZIMMERMANN, 2009; GAVRAS et al., 2007).

Esse trabalho consiste em comparar duas arquiteturas de rede *clean slate* a *Entity Title Architecture*, ou Arquitetura de Entidade e Título (ETArch) e a *Content Centric Networking*, ou Rede Centrada em Conteúdo (CCN). Essas duas arquiteturas fazem parte

das Information-Centric Networking (ICN) que marca uma mudança significativa na forma de comunicação. As ICNs são capazes de suportar vários requisitos de rede do futuro, entretanto a comparação entre essas diferentes arquiteturas de forma igualitária não é trivial. A avaliação de desempenho de um protocolo bem estabelecido, como o TCP, têm o benefício de representar uma carga de trabalho realista e reflete um ambiente onde os serviços e os protocolos serão implantados. No entanto, os resultados obtidos neste ambiente são muitas vezes difíceis de replicar de forma independente o que afeta os resultados experimentais entre redes ICNs (LABOVITZ et al., 2010; FARIAS et al., 2011).

A arquitetura ETArch é uma instância do *Entity Title Model* ou Modelo de Entidade e Título (PEREIRA et al., 2011). Alguns conceitos básicos dessa arquitetura são *Domain Title Service*, ou Serviço de Domínio de Títulos (DTS), a Entidade, o Título e o *Workspace*. Uma Entidade tem a necessidade de se comunicar e pode ser caracterizada por: uma aplicação, um *smartphone*, um *host*, entre outros. O Título identifica uma Entidade de forma única, não ambígua e independe da topologia. Já o *Workspace* é um meio pelo qual as Entidades são capazes de se ligar para participar de uma comunicação. A iniciação, manutenção e encerramento de um *Workspace* é controlado por um sistema de títulos denominado de DTS (SILVA, 2013).

Com a abordagem *clean slate* ETArch é possível suportar *multicast* que permite a entregar dados para múltiplos destinatários simultaneamente usando uma estratégia mais eficiente para evitar a replicação de primitivas. Além de garantir mobilidade e transmissão de um stream sem interrupção tanto para redes cabeadas como sem fio (SILVA, 2013).

A abordagem CCN consiste em enfatizar o *content* (dados ou informações) tornando-o diretamente endereçável e roteável. Diferentemente do modelo atual de Internet a comunicação *endpoint* nessa arquitetura, não é baseada no endereço IP mas sim no nome dos dados. A CCN é baseada na troca de mensagens *request content* (chamada de *Interests*) e as mensagens *return content* (chamadas de *Content Objects*). Além disso os nomes e os *contents* são armazenados automaticamente em *cashes* que são blocos de memória usados para armazenamento de dados temporários de acesso rápido. As *cashes* são distribuídas em roteadores de forma a otimizar o tráfego de dados da rede (KOPONEN et al., 2007; PERINO; VARVELLO, 2011).

A arquitetura *clean slate* CCN oferece uma rede mais segura, flexível e escalável atendendo assim às exigências modernas da Internet para a distribuição de conteúdo protegido e em grande escala a um conjunto diversificado de dispositivos finais. Além disso, quando solicitado um pedido, essa arquitetura chama o *content* para o usuário a partir do

cache mais próximo, atravessando menos saltos de rede, eliminando pedidos redundantes e consumindo menos recursos em geral (KOPONEN et al., 2007; OEHLMANN, 2013).

No decorrer deste capítulo será apresentado os objetivos desse estudo, na seção 1.1. Logo em seguida será mostrado algumas razões para a realização desse trabalho na seção 1.2. E depois será apresentado, na seção 1.3, um breve resumo de pesquisas envolvendo as redes de Internet do Futuro.

1.1 Objetivos

O objetivo geral desse projeto é fazer uma análise comparativa entre as abordagens *clean slate* de Internet do Futuro CCN e ETArch. As arquiteturas CCN e ETArch serão implantadas de modo a compará-las usando uma aplicação de *Chat*.

Os objetivos específicos desse trabalho será mostrado a seguir.

- Definir um conjunto de critérios adequados a fim de comparar as abordagens CCN e ETArch;
- Pesquisar e implantar a abordagem CCN para Internet do Futuro, a fim de realizar sua avaliação experimental no caso de uso definido para esse trabalho;
- Pesquisar e implantar a abordagem ETArch para Internet do Futuro, a fim de realizar sua avaliação experimental no caso de uso definido para esse trabalho;
- Comparar as abordagens CCN e ETArch baseado nos critérios definidos anteriormente.

Ao final desse trabalho será apresentado, de acordo com os conjuntos de critérios definidos, uma análise comprativa entre as abordagens CCN e ETArch.

1.2 Justificativa

No cenário atual da Internet, temos um conjunto de necessidades vinda de sucessíveis evoluções no modelo TCP/IP. Algumas evoluções no modelo TCP/IP são *Classless Inter-Domain Routing* (CIDR) (FULLER; LI, 2006), *Network Address Translation* (NAT)

(SRISURESH; HOLDREGE, 1999), alocação de endereços IPv4, IPv6 e a tabela de roteamento BGP (MENG et al., 2005; FARIAS et al., 2011). No intuito de sanar as novas necessidades da rede, a comunidade de pesquisa de Internet do Futuro vêm estudando algumas abordagens *clean slate*.

O estudo comparativo experimental das arquiteturas CCN e ETArch permite avaliar, na prática, duas arquiteturas *clean slate* que possibilita aos pesquisadores reforçar os aspectos centrais de cada uma dessas abordagens. Além disso permite uma comparação direta, entre essas abordagens, ajudando a comunidade de pesquisa a identificar e propor novas melhorias tanto para as redes de Internet do Futuro como para as ferramentas usadas para implementá-las.

Com esse estudo os pesquisadores envolvidos com Internet do Futuro terão mais informações para analisar e comparar as arquiteturas CCN e ETArch com as outras abordagens *clean slate*. E através dos testes e levantamentos, feitos nessa pesquisa, gerar discussões com a proposta de gerar um conceito de rede mais eficiente, eficaz e capaz de atender os novos requisitos da Internet.

1.3 Estado da Arte

Diversos grupos de pesquisa estão envolvidos em projetos cujo foco são novas arquiteturas de rede que atendem aos requisitos da Internet do Futuro. Muitos pesquisadores (PEREIRA et al., 2011; SILVA, 2013; CASTILLO et al., 2014) fizeram estudos envolvendo a arquitetura ETArch e outros (HONG; JANG; LEE, 2013; AMBIEL; MAGALHÃES, 2013; DETTI et al., 2011) a CCN procurando atender alguns requisitos da rede do futuro, como: *multicast* ou capacidade de entregar dados para múltiplos destinatários simultaneamente de forma eficiente, *multihoming* ou técnica capaz de utilizar múltiplos pontos de conexão e *QoS/QoE*, ou seja, garantir qualidade de serviços oferecendo uma rede com alta disponibilidade e com menos atrasos de pacotes, assim como qualidade de experiência oferecendo segurança, fidelidade dos dados e tempo de resposta rápido.

Os pesquisadores envolvidos com o desenvolvimento de arquiteturas *clean slate* estão adotando, nos últimos anos, um novo paradigma para as redes chamado de *Software Defined Networking* (SDN). Um dos principais conceitos das arquiteturas SDNs é separar o plano de dados do plano de controle permitindo que diferentes tipos de *hardwares* possam ser controlados por aplicações usando interfaces padrões. Além disso esse novo paradigma permite construir uma rede altamente escalável, flexível e controlada (BRAUN; MENTH, 2014).

O *OpenFlow* é uma tecnologia que permite materializar os conceitos envolvendo as redes SDNs. Na Europa, o projeto OFELIA, construiu uma instalação experimental baseada em *OpenFlow* que interconecta oito ilhas em diferentes países (SUñé et al., 2014). No Brasil o projeto FIBRE (FIBRE, 2013) tem por objetivo interconectar diferentes instituições de ensino. Esses estudos permitem aos pesquisadores a criação de um ambiente mais próximo do real aumentando a confiabilidade dos experimentos, conforme necessário para arquiteturas como a ETArch e a CCN (SILVA, 2013).

Segundo a tese de (SILVA, 2013), a arquitetura ETArch estabelece novas relações entre as entidades de sistemas distribuídos, e implementa uma definição de *Workspace*. O *Workspace* permite estabelecer um canal por onde vários nós podem se comunicar de forma *multicast*, além de permitir a mobilidade dos nós ao longo da rede.

O estudo de (HONG; JANG; LEE, 2013) envolvendo a abordagem CCN para aplicativos *mobile* mostra que é possível reduzir a quantidade de tráfego usando o armazenamento em rede e fornecer segurança. De acordo com o artigo (HONG; JANG; LEE, 2013) a abordagem CCN é muito flexível e com a utilização dos campos de operação pode se estender a capacidade dessa arquitetura.

A seguir temos os seguintes capítulos: O capítulo 2 define a proposta de avaliação comparativa entre as abordagens *CCN* e *ETArch*. No capítulo 3 é feita uma contextualização dos conceitos e tecnologias envolvidas nesse trabalho. Então no capítulo 4 é detalhado os experimentos realizados nesse estudo assim como os resultados coletados. E por fim no capítulo 5 é feita uma conclusão a respeito deste trabalho.

2 Método

A abordagem *clean slate* tem sido bem aceita pelos pesquisadores envolvidos, surgindo assim a necessidade de um processo de comparação envolvendo a redes de internet no futuro. A metodologia de avaliação para arquiteturas *Information-Centric Networking* (ICN) tem a proposta de sugerir um conjunto de dados e abordagens de alto nível que serão capazes de comparar diferentes redes ICN. Como as arquiteturas ICN é uma área em crescimento as ferramentas de avaliação ainda estão em processo de desenvolvimento, e ainda não se tem um ambiente capaz de suportar as abordagens ICN mais conhecidas (PENTIKOUSIS B. OHLMAN, 2016).

Com isso diferentes tipos de métodos comparativos estão sendo avaliados pela comunidade ICN, como análise teórica, simuladores, emuladores, *testbeds* e até a preparação de ambientes locais executáveis. A escolha do método a ser usado depende do objetivo da pesquisa em questão, e se será avaliado escalabilidade, quantidade de recursos utilizados ou análise de incentivos econômicos (PENTIKOUSIS B. OHLMAN, 2016).

Sendo assim, o primeiro passo a ser definido neste trabalho para a comparação entre as abordagens CCN e ETArch é a definição de qual método comparativo será usado.

Das várias topologias de rede usadas em estudos envolvendo o paradigma ICN não há uma topologia que avalie todos os aspectos de uma arquitetura *clean slate*. Então, o segundo passo desse projeto é definir uma topologia que será usada para a comparação das abordagens CCN e ETArch (PENTIKOUSIS B. OHLMAN, 2016).

Os testes comparativos dependem do tipo da aplicação e de um conjunto de métricas a serem avaliadas na análise. Portanto o terceiro passo é a definição do conjunto de critérios para comparação das arquiteturas e da escolha de um tipo de aplicação (PENTIKOUSIS B. OHLMAN, 2016).

O quarto passo será implantar as arquiteturas CCN e ETArch usando o método de comparação escolhido. A ideia nesse passo é pegar uma implementação de cada abordagem (CCN e ETArch) e colocá-la em funcionamento no ambiente definido.

O último passo é a realização dos testes, análise e conclusão levando se em consideração as definições feitas nos passos anteriores.

Na sequência é apresentado o próximo capítulo envolvendo uma contextualização da Internet e a sua atual condição apresentando, nesse caminho, algumas tecnologias pontuais para chegar no que temos hoje envolvendo a Internet do Futuro.

3 Referencial Teórico

Nesse capítulo é feita uma introdução do surgimento da internet e o que levou a rede das redes a chegar no que temos hoje, seção 3.1. Depois na seção 3.2 é apresentado a tecnologia *openflow* e um novo conceito de rede. Na seção 3.3, é conceituado a abordagem *CCN*. Já na seção 3.4 é feito a definição da arquitetura *Etarch*. Por fim a seção 3.5 apresenta algumas pesquisas relacionadas a esse estudo.

3.1 Internet do Futuro

A Internet teve sua origem na década de 70 e foi criada com a proposta de atender uma rede entre universidades onde os usuários eram confiáveis e tinham conhecimentos técnicos sobre a rede. Nesta época alguns dos requisitos para a Internet era conectividade, robustez, heterogeneidade, acessibilidade e responsabilização. Para atender esses requisitos, adotou-se a utilização de um modelo em camadas, o TCP/IP (MOREIRA et al., 2009).

- Conectividade: Possibilitar o tráfego de dados entre as redes;
- Robustez: Permitir a comunicação de dados por um caminho entre um ponto de origem e um de destino ;
- Heterogeneidade: Suportar diferente tipos de tecnologias de rede assim como aplicações e serviços;
- Acessibilidade: Facilitar a adição de novos nós na rede;
- Responsabilização: Possibilitar a identificação de recursos na Internet;

O modelo TCP/IP permitiu a simplificação na criação de novas aplicações, já que não havia a necessidade de modificar a estrutura interna da rede. Com isso houve um crescimento do uso e de maneira diversificada da Internet. A facilidade de criar novas aplicações e serviços juntamente com o surgimento de novos requisitos fez-se necessário a inclusão de novas extensões na estrutura inicial da rede. As extensões mais conhecidas foram *Classless Internet Domain Routing* (CIDR), *Network Address Translation* (NAT),

Serviços Integrados (Intserv), Serviços Diferenciados (Diffserv), IP Seguro, IP Móvel e *firewalls* (FARIAS et al., 2011).

Nesse contexto temos a arquitetura de Internet atual com as limitações e problemas que esse processo de criação e modificação gerou. Sendo assim temos a necessidade de resolver os novos requisitos de rede como endereçamento, mobilidade, segurança, confiabilidade da rede, disponibilidade de serviço, diagnóstico de problemas, gerenciamento de rede, qualidade de serviço, qualidade de experiência e escalabilidade (MOREIRA et al., 2009).

- Endereçamento: Criar novas formas de identificar e localizar um nó na rede além de pensar em definir um espaço de endereçamento compatível com a realidade atual;
- Mobilidade: Permitir a movimentação do nó entre diversos pontos de acesso mantendo as conexões ativas;
- Segurança: Criar uma arquitetura de segurança capaz de gerar um ambiente confiável;
- Confiabilidade da rede e Disponibilidade de serviço: Disponibilizar um serviço de rede com alta confiabilidade e que esteja sempre disponível;
- Diagnóstico de problemas e Gerenciamento de rede: Criar ferramentas capazes de diagnosticar os problemas da rede assim como permitir o gerenciamento da rede;
- Qualidade de Serviço e Qualidade de Experiência: Criar novos métodos de encaminhamento de pacote baseado em cada aplicação(reservar recursos, definir prioridade e etc) garantindo qualidade da aplicação na rede e qualidade de experiência do usuário;
- Escalabilidade: Suportar o aumento exponencial na quantidade de usuários da Internet mantendo o sistema de roteamento global escalável;

A Internet do futuro consiste, principalmente, em criar e avaliar soluções para Internet que seja capaz de atender os novos requisitos de rede. Nesse caminho os pesquisadores estão divididos em duas abordagens para solucionar as novas necessidades da rede. Uma das abordagens é conhecida como *clean slate* que visa substituir a arquitetura atual por uma completamente nova enquanto que a outra, *Evolutionary*, pretende evoluir a arquitetura atual(TCP/IP).

Diante de algumas limitações da arquitetura tradicional como exaustão de identificadores de rede (IPv4), custo elevado de roteadores IP, tabelas de roteamento não escalonável, problemas de segurança (*denial of service* (DoS), spam e crimes de roubo de informação) além da dificuldade de combinar transparência de acesso e desempenho de aplicação para usuários móveis os pesquisadores estão tendendo a soluções que envolvem arquiteturas *clean slate*.

A comunidade envolvida com Internet do Futuro encontrou uma barreira que tem sido a capacidade de experimentar as novas propostas de forma realista, sem interferir na base de equipamentos e protocolos usados na rede atual. Tendo reconhecido esse fato, os pesquisadores começaram a desenvolver alternativas para experimentação capaz de replicar um cenário mais realista.

Atualmente temos vários programas de pesquisa para experimentação oferecendo infraestruturas capazes de habilitar, controlar e testar arquiteturas *clean slates*, de forma a replicar um ambiente realista capaz de validar eficientemente essas abordagens e sem prejudicar a estrutura atual da rede (GUIMARAES et al., 2014; GAVRAS et al., 2007).

3.2 *OpenFlow* e *Software Defined Networking* (SDN)

A tecnologia *OpenFlow* é uma solução que vem oferecendo aos pesquisadores a possibilidade de testar seus protocolos experimentais em redes mais realistas como: redes de um campus, redes metropolitanas e até redes envolvendo alguns países. O *OpenFlow*, além de oferecer o protocolo de controle, para manipular a tabela de encaminhamento dos roteadores e switches, também oferece uma API (Application Programming Interface) simples e extensível para programar o comportamento dos fluxos de pacotes. Através desta API, os pesquisadores podem rapidamente construir novos protocolos e aplicá-los em um ambiente de produção sem comprometê-lo (BRAUN; MENTH, 2014).

A Figura 1 representa a arquitetura da tecnologia *OpenFlow* contendo o *Openflow Controller* e a especificação do *Openflow Switch*. O *Openflow Controller* usa o protocolo *OpenFlow* para conectar e configurar dispositivos de redes (roteadores, *switches*, etc.) para determinar o melhor caminho para o tráfego de dados de uma determinada aplicação. O *Openflow Switch* é um dispositivo de *hardware* capaz de encaminhar os pacotes do controlador *OpenFlow*.

Com o *OpenFlow* e os protocolos de controle foi possível materializar o conceito de SDN *Software Defined Networking*, ou Redes Definidas por Software que essencialmente

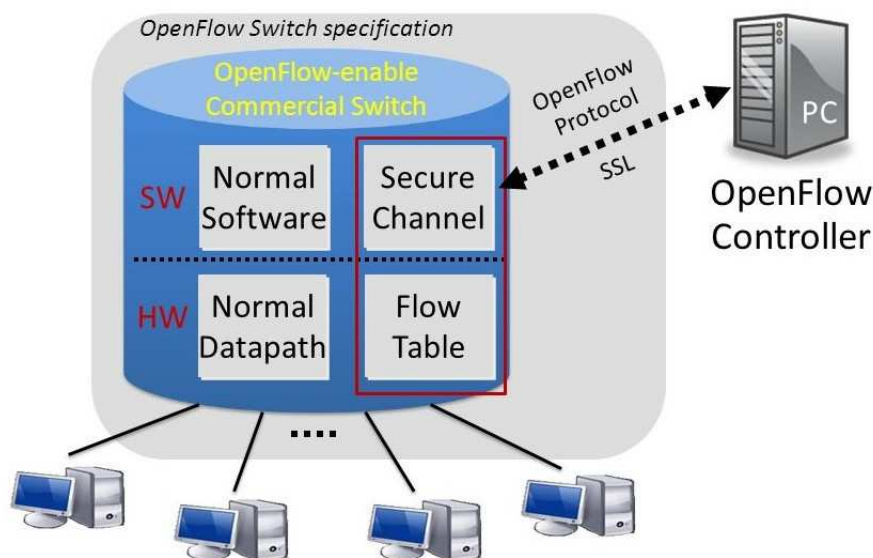


Figura 1 – Exemplo da arquitetura *OpenFlow* (OPENFLOW, 2010)

consiste em separar o plano de dados do plano de controle, permitindo criar uma rede altamente escalável, flexível e controlada. (FERREIRA et al., 2014).

A Figura 2 representa o conceito da rede SDN que busca centralizar o controle da rede, separando a lógica de controle para recursos de rede fora do dispositivo. Todos os modelos SDN possuem um *Controller*, *Southbound APIs* e *Northbound APIs*. Os *Controllers*, ou os Controladores SDN oferecem uma visão centralizada da rede geral e permitem que os administradores de rede ditem aos sistemas subjacentes (como *switches* e roteadores) como o plano de encaminhamento deve lidar com o tráfego da rede. As *APIs Southbound*, ou APIs da parte de "baixo", retransmitem informações para os switches e roteadores. O *OpenFlow* é considerado o primeiro padrão da rede SDN como *APIs Southbound*. As *APIs Northbound*, ou APIs da parte de "cima", se comunicam com as aplicações e a lógica de negócios "acima". Isso ajuda os administradores de rede a configurarem sistematicamente o tráfego e implantar serviços.

Pórem a comunidade SDN enfrenta atualmente um cenário desafiador onde os controladores tem que atender os requisitos de alta disponibilidade, escalabilidade, alto desempenho, confiabilidade, tolerância a falhas, capacidade de gerenciamento além de permitir a modularização de novos serviços independentemente dos dispositivos da rede (SILVA et al., 2014b). No cenário atual os pesquisadores estão divididos em duas abordagens. A primeira *bottom-up* consiste em construir uma camada de controle SDN do zero. A outra *top-down*, consiste em manter um modelo de componente, que já está sendo usado, integrando módulos para oferecer novos tipos de serviço (AHOKAS, 2014).

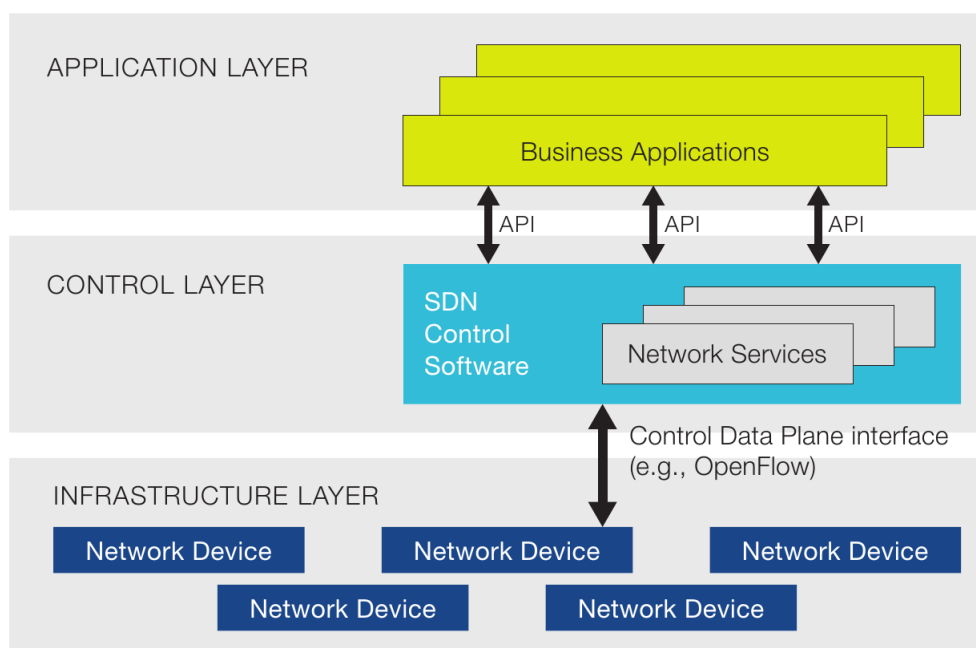


Figura 2 – Exemplo da arquitetura SDN (SDXCENTRAL, 2015)

No trabalho de (SILVA et al., 2014b) é proposto um novo controlador baseado na arquitetura SDN usando uma abordagem *top-down*. No estudo, (SILVA et al., 2014b), usou um modelo de componente especializados para o desenvolvimento de aplicações baseadas em eventos JAIN SLEE (*Java APIs for Integrated Networks - Service Logic Execution Environment*) e integrado a ele um adaptador *OpenFlow* e um adaptador MIH (*Media Independent Handover Services*). Com isso foi possível criar um controlador que suporta tolerância a falha, controla a vazão e os atrasos da rede além de ser capaz de lidar com diferentes protocolos onde novos serviços e aplicações pode ser integrados.

Atualmente dois grandes trabalhos usam a abordagem *bottom-up* para construir um controlador *OpenFlow*. A primeira é chamada *OpenDaylight* que é uma plataforma aberta que oferece um ambiente para programação de rede SDNs de qualquer tamanho e escala. A segunda é conhecida como *Open Network Operating System (ONOS)* que também permite a criação de redes SDNs dando suporte a disponibilidade, performance e escalabilidade.

3.3 Content Centric Networking (CCN) / Named Data Networking (NDN)

A Internet atualmente tem sido usada largamente para a distribuição de conteúdos complexos. Isso devido ao crescimento no uso de aplicações que envolvem vídeo conferência, jogos em rede e mensagens instantâneas que além de ter restrições de tempo para execução pode fazer a interação de vários usuários em tempo real. Essas aplicações criaram a necessidade de se ter qualidade, eficiência e confiabilidade na entrega de conteúdos (AMBIEL; MAGALHÃES, 2013).

No modelo da internet atual(TCP/IP) a comunicação é feita através de um canal entre duas pontas identificadas pelo IP, através dos pacotes IP, ou *IP Packets*. Enquanto que na abordagem CCN o principal meio de comunicação são os blocos de conteúdos, ou *Content Chunks*. A Figura 3 representa as pilhas de protocolos envolvendo as arquiteturas TCP/IP e CCN.

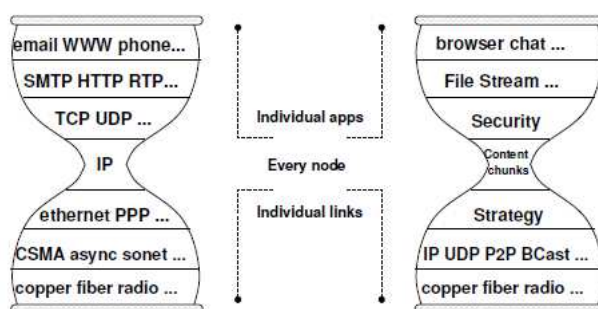


Figura 3 – Exemplo de um nó CCN(a direita) e um nó da Internet atual (a esquerda) (KANG BYUNGRYEOL SIM; LEE, 2012)

As Redes Orientadas a Conteúdo (ROCs) tem como foco o armazenamento e a distribuição de conteúdos. As ROCs procuram de forma eficiente garantir disponibilidade e tem como princípio o nome dos dados para solucionar a semântica de localização. A *Content Centric Networking* (CCN) ou *Named Data Networking* (NDN) vem como uma proposta das ROCs e tem uma forma hierárquica e inteligente para nomear os conteúdos. Como podemos ver na Figura 4, temos o exemplo de uma estrutura para nomeação de um conteúdo nas redes CCNs que é similar aos nomes de domínios (CABRAL; ROTHENBERG, 2013).

Apenas com o nome dos dados, na rede CCN, já é possível ter informações a respeito do formato do arquivo e de sua versão. O esquema de nomeação permite também a

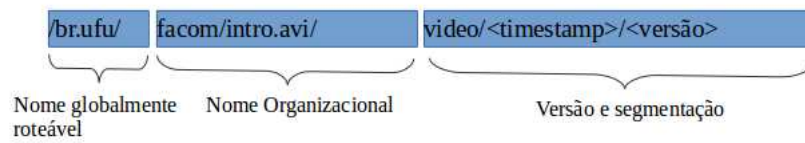


Figura 4 – Exemplo de nome de um conteúdo na rede CCN

adição de novos prefixos trazendo assim uma maior escalabilidade do roteamento uma vez que os próprios nomes na arquitetura podem ser usados com a finalidade de roteamento.

A abordagem CCN conta com dois tipos de pacotes: *Interest* e *Data* ou *Content Object*, Figura 5. Todos os nós na arquitetura CCN pode servir como um provedor ou consumidor de conteúdo. A comunicação é iniciada quando um nó consumidor requisita um conteúdo enviando um pacote *Interest* pelas conexões disponíveis. Assim que essa mensagem chega a um nó produtor (que contém o conteúdo requisitado), a mensagem é então consumida criando um pacote *Data* contendo o conteúdo requerido pela mensagem e enviado para o nó consumidor.

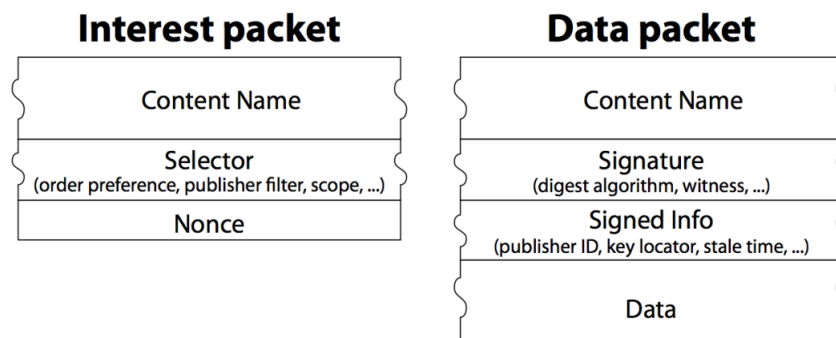


Figura 5 – Exemplo dos dois tipos de pacote da rede CCN: *Interest* e *Data* (CABRAL; ROTHENBERG, 2013)

Os nós CCN possuem três estruturas de dados principais (Figura 6): A *Forward Information Base* (FIB) é uma tabela de encaminhamento, que armazena as interfaces por onde um pacote *Interest* pode ser enviado; A *Pending Interest Table* (PIT) uma tabela de interesse onde são armazenados as interfaces de recepção e o nome do conteúdo e o *Content Store* (CS) que é uma estrutura que permite salvar pacotes *Data*.

Cada nó CCN pode armazenar temporariamente pacotes *Data* no CS. Essa estrutura permite a criação de um cache interno já que cada pacote *Data* que chega ao nó é armazenado pelo maior tempo possível aumentando assim a probabilidade de um mesmo conteúdo ser amplamente compartilhado a medida que é consumido por outro nós. Com

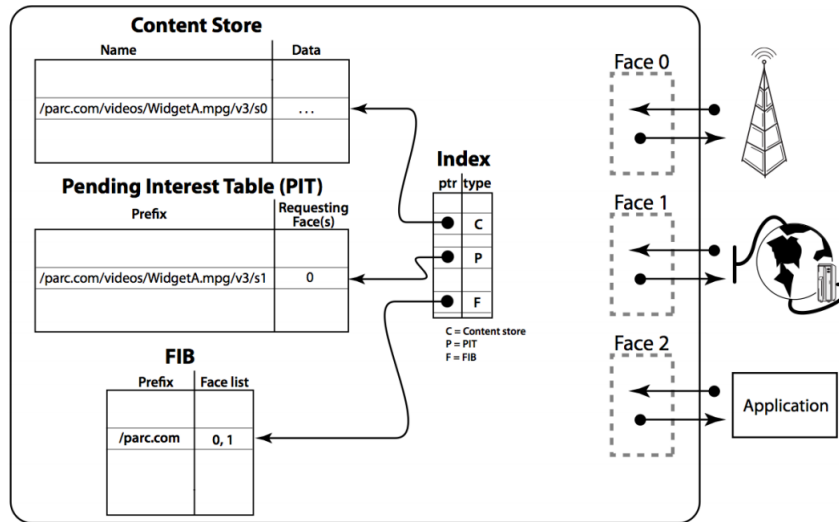


Figura 6 – Exemplo das estruturas de um nó CCN (AMBIEL; MAGALHÃES, 2013)

esse processo as redes CCNs buscam minimizar o consumo de banda e prover uma rede eficiente.

Quando um pacote *Interest* passa por um nó, esse verifica se o conteúdo de interesse está salvo no CS. Se o conteúdo for encontrado, é criado um pacote *Data* com o dado requisitado e imediatamente enviado ao nó que originou a requisição. Se não for encontrado, o nome do conteúdo procurado é juntamente com a interface de recepção do pacote inserido na tabela de interesse (PIT). Assim a PIT armazena todos os *Interests* que passam por um nó em direção ao produtor do conteúdo. Com isso quando o pacote *Data* voltar ele vai ser direcionado para o(s) consumidor(es) pelo mesmo caminho seguido ao procurar o produtor, removendo da tabela de interesse dos nós as entradas referentes ao conteúdo buscado. No caso de não encontrar nenhum nó com o conteúdo, eventualmente as entradas na PIT sofrem *timeout*.

Toda vez que uma entrada é adicionada na tabela de interesse, o pacote *Interest* é enviado para a tabela de encaminhamento (FIB) para encontrar uma interface que bata com o prefixo do nome do conteúdo mais longo. Caso exista pelo menos uma entrada que satisfaça a busca, o pacote *Interest* é encaminhado pela interface correspondente, caso contrário o pacote é descartado.

3.4 ETArch

A arquitetura ETArch é uma abordagem *clean slate* que foi criada baseada nos conceitos do modelo *Entity Title Model* e das redes *Software Defined Networking* (SDN).

O Entity Title Model define quais são os requisitos e capacidades das entidades para que elas possam se comunicar. A proposta ETArch propõem uma nova forma de identificação e endereçamento além de permitir a mobilidade dos nós e a capacidade de realizar *multicast*. Alguns conceitos básicos dessa arquitetura de Internet do Futuro são: *Entity*, *Title*, *Workspace* e *DTS*. Além dos componentes da arquitetura ETArch, a Figura 7, apresenta os principais protocolos envolvidos no plano de controle e no plano de dados (SILVA et al., 2014b; SILVA et al., 2014a; SILVA, 2013).

A Entidade (*Entity*) é representada por um Título (*Title*), pela qual ela é unicamente identificada, e uma localização definida como Ponto de Conexão (*Point of Attachment* (PoA)). Uma Entidade pode se comunicar com outras e até compartilhar propriedades, com exceção do título. A grande vantagem da separação entre identificação e localização nas Entidades é a possibilidade de mobilidade destas na rede, já que a localização pode mudar enquanto a identificação permanece a mesma, o que não é possível na rede atual (SILVA, 2013; LEMA, 2014).

A definição das Entidades na rede ETArch trás uma grande flexibilidade que permite atender diferentes requisitos para diferentes abordagens como CCN (*Content Centric*), TCP/IP (*Host Centric*), dispositivos em geral (*Internet of Things* (IoT)) entre outras (SILVA, 2013; LEMA, 2014).

O Título (*Title*) é usado para identificar uma Entidade. Para atender esse propósito, o Título, deve ser único independente de qualquer topologia. Além disso, o Título deve referenciar uma única Entidade enquanto que uma Entidade pode ter mais de um Título. O Título tem um papel importante para garantir o endereçamento das Entidades (SILVA, 2013; LEMA, 2014).

Na arquitetura ETArch a comunicação de uma Entidade com a(s) outra(s) é feita através de um canal, chamado *Workspace*. Quando uma Entidade necessita de uma informação, é criada uma mensagem e enviada para o *Workspace* que, por sua vez, se encarrega de entregá-la a(s) Entidade(s) de destino. O *Workspace* permite o uso eficiente da camada física e abstrai a tecnologia de rede envolvida em cada uma das Entidades (SILVA, 2013).

Para suportar o conceito do *Workspace* criou-se, no plano de controle, o *Domain Title Service* (DTS). O DTS através dos *Domain Title Service Agent* (DTSA) é responsável pelo controle de toda rede ETArch permitindo serviços como: criação, atualização e remoção de um *Workspace* e associação e desassociação de Entidades de um determinado *Workspace* (SILVA, 2013).

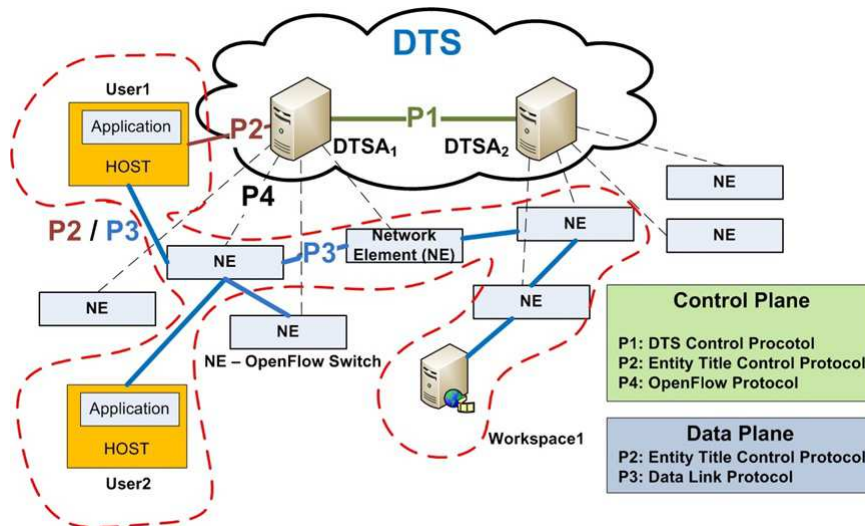


Figura 7 – Exemplo dos Principais Componentes e Protocolos da rede ETArch (SILVA, 2013)

A ETArch define uma nova pilha de protocolo. Como podemos ver na figura 8, a pilha de protocolo ETArch, define uma nova camada *Communication Layer*. Essa camada de comunicação abstrai algumas funcionalidades da camada de transporte e de rede do modelo OSI. A camada de aplicação (*Application Layer*) suporta os protocolos usados no modelo OSI porém não será limitada a esses protocolos. Na camada de enlace (*Link Layer*) será definido um novo *header* definindo agora o endereçamento para Título de uma Entidade. Com poucas mudanças é possível tornar a ETArch compatível com a internet atual (CASTILLO et al., 2014).

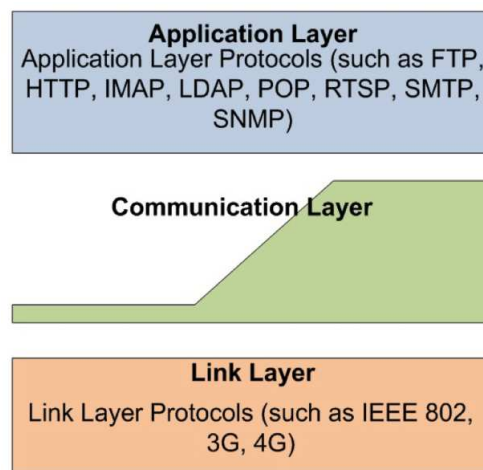


Figura 8 – Exemplo dos Pilha de Protocolo da rede ETArch (CASTILLO et al., 2014)

Na figura 9 temos uma Entidade disponibilizando um dado na arquitetura ETArch. Com esse intuito a Entidade faz uma requisição para o DTSA pelo *switch OpenFlow*. O DTSA irá receber essa mensagem e adicionará uma regra na *flow table*, associando

a entidade geradora da informação ao novo *Workspace*. Qualquer Entidade que quiser receber esse dado, irá enviar uma requisição para o DTSA para se conectar (*Attach*) ao *Workspace* criado. O DTSA, por sua vez, irá modificar a *flow table* incluindo a Entidade que está querendo receber a informação no *Workspace* (LEMA, 2014).

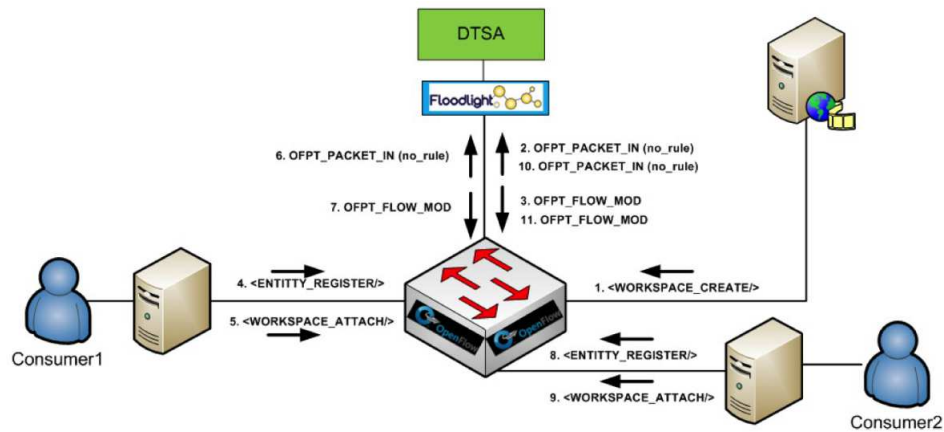


Figura 9 – Exemplo de iniciação de um *Workflow* na rede ETArch (CASTILLO et al., 2014)

3.5 Trabalhos Correlatos

Existe na literatura diversos trabalhos que tratam da comparação entre arquiteturas *clean slate*, em especial as que utilizam o paradigma ICN, entretanto esses trabalhos não foram realizados de forma experimental.

Os trabalhos de (AHLGREN et al., 2012; XYLOMENOS et al., 2014) são feitas várias comparações e discussões a respeito de difentes abordagens ICNs. Entretanto, os estudos de (AHLGREN et al., 2012; XYLOMENOS et al., 2014) apresentam apenas comparações qualitativas enquanto que esse trabalho basea-se em uma comparação quantitativa baseada em métricas bem definidas.

A pesquisa de (LI et al., 2014) faz a comparação entre duas arquiteturas ICNs, *Mobility First* (MF) e NDN(CCN). Na análise comparativa de (LI et al., 2014), as duas arquiteturas *clean slate*, tem o mesmo paradigma(orientada a conteúdo) além de se considerar um modelo *pub/sub*(esse modelo consiste em descentralizar a disponibilidade de conteúdo e recurso da arquitetura). Por outro lado a proposta desse estudo é analisar os aspectos envolvendo performance da rede(tamanho de pacotes e *throughput*) sobre dois paradigmas distintos CCN(orientada a conteúdo) e ETArch(orientada a *workspace*).

A comparação experimental no trabalho de (SILVA, 2013) não envolve nenhuma rede ICN, já que seu trabalho leva em consideração apenas as arquiteturas EtArch e TPC/IP. Porém nesse estudo o processo comparativo é feito entre duas abordagens distintas, uma orientada a *workflow*(ETArch) e outra a conteúdo(CCN) está última por sua vez faz parte das ICNs. Sendo assim, essa pesquisa, define um conjunto de critérios a serem avaliados por dois paradigmas diferentes que busca ajudar e contribuir para uma rede com maior qualidade de serviço(QoS) capaz de garantir largura de banda e eficiência na hora de trafegar pacotes na rede.

No próximo capítulo temos os experimentos realizado nesse trabalho assim como os resultados obtidos na comparação das arquiteturas *clean slate* CCN e ETArch.

4 Avaliação Experimental

Inicialmente este capítulo apresentará a metodologia de avaliação para as arquiteturas avaliadas nesse estudo na seção 4.1. Em seguida será apresentado a implementação das arquiteturas CCN e ETArch nas seções 4.2 e 4.3, respectivamente. E finalizando o capítulo será apresentado, na seção 4.4, os experimentos realizados nessa pesquisa.

4.1 Metodologia

A metodologia de comparação das arquiteturas de Internet do Futuro abordadas nesse trabalho seguirá os passos definidos no capítulo 2. Sendo assim, como primeiro passo da metodologia, o método comparativo a ser usado vai ser a preparação de um ambiente local executável. Esse ambiente local é uma máquina de uso pessoal equipada com 4 GB de memória RAM, e com processador Intel Core i3 4 x 1.33 GHz com sistema operacional linux 14.04 LTS 64-bit. Nesse ambiente foi instalado as ferramentas apresentadas na tabela 1.

Tabela 1 – Tabela com as ferramentas usadas no experimento

Ferramenta	Descrição	Versão
<i>Virtual Box</i>	Permite a virtualização de aplicativos.	5.0
<i>Etarch</i>	Permite a representação da rede ETArch e seus componentes.	1.0.0
<i>Mininet</i>	Permite a simulação de diferentes <i>host</i> , entidades, <i>switches</i> e suas ligações.	2.1.0
<i>Mini-CCNx</i>	Permite a representação da rede CCN e seus componentes.	0.8.2
<i>Wireshark</i>	Permite a captura de tráfego de dados da rede.	1.12.1

Como segundo passo a topologia a ser usada nos experimentos será uma topologia *full mesh* contendo três nós como podemos ver na Figura 10.

A metodologia de avaliação (PENTIKOUSIS B. OHLMAN, 2016) sugere alguns conjuntos de dados para comparar diferentes tipos de redes ICNs. De acordo com essa metodologia, para a comparação das arquiteturas CCN e ETArch, foi definido os seguintes conjuntos de métricas(passo três):

- *Throughput* de rede: é a quantidade de dados processados em um determinado espaço de tempo;

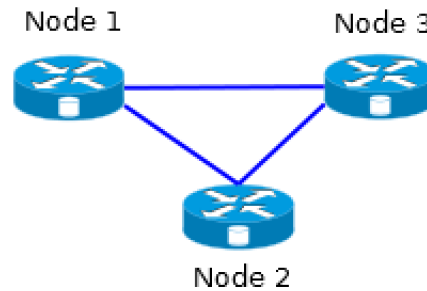


Figura 10 – Exemplo da topologia *full mesh* usada para os testes

- *Packet length*: é o tamanho do pacote de dados;

Além do conjunto de métricas definido foi escolhido o *Chat* como aplicação para o experimento. Uma vez que o *Chat* é uma aplicação usada em larga escala constantemente.

O quarto passo consiste na implantação das arquiteturas CCN e ETArch que são apresentadas nas seções 4.2 e 4.3, respectivamente.

E por último é apresentado os testes e análise realizados nesse estudo (4.4).

4.2 Implementação: CCN

Para a implementação da arquitetura CCN foi utilizado a ferramenta *Mini-CCNx* (versão 0.8.2) que vem sendo trabalhada desde 2010 pelo *Palo Alto Research Center* (PARC) (ROTHENBERG, 2013). Cada nó do *Mini-CCNx* emula um nó do CCN rodando exatamente o código oficial CCN. A emulação é feita usando uma técnica de virtualização *lightweigh* denominada *Container-Based Emulation* (CBE). Em que cada container (nó CCN) permite que grupos de processos tenham visões independentes dos recursos do sistema como IDs de processo, sistema de arquivo e interfaces de rede usando o mesmo kernel. Além disso, cada container tem sua própria interface de rede, interfaces de redes virtuais, o *daemon ccnd* (que implementa as estruturas CS, FIB e PIT) e os repositórios para conteúdos (*ccnr*). E cada nó (container) são conectados entre si usando um *link* virtual *Ethernet* (CABRAL; ROTHENBERG; MAGALHÃES, 2013).

O *Mini-CCNx* permitiu criar vários nós CCN com agilidade e flexibilidade e configurar diferentes tipos de topologias. Além de garantir confiabilidade nos resultados dos testes, uma vez que, o *Mini-CCNx* tem tecnologias de limitação para controlar os recursos de cada nó e de cada *link*.

4.3 Implementação: ETArch

A implementação da abordagem ETArch foi feita configurando duas máquinas virtuais usando o *Virtual Box* (versão 5.0) (VIRTUALBOX, 2017). O *Virtual Box* é uma ferramenta usada para fazer a virtualização de aplicações (ROMERO, 2010). Nesse estudo foi feita a virtualização do ETArch (versão 1.0.0) uma máquina pronta com todos os componentes definidos na seção 4.3 da rede ETArch. E a segunda máquina virtual usada foi a *Mininet* (versão 2.1.0) (ETARCH, 2017).

A *Mininet* é uma boa maneira de desenvolver, compartilhar e experimentar sistemas como *OpenFlow* e *SDN*. O *Mininet* é um sistema de prototipagem rápida para redes que simula diferentes *host*, entidades, *switches* e suas ligações. Essa ferramenta possui uma *Command Line Interface* (CLI) que permite a interação com diferentes *hosts* (Entidades) e uma *Application Programming Interface* (API), baseada em Python, que viabiliza a criação de diversas topologias (SILVA, 2013).

4.4 Experimentos e Resultados

Os experimentos realizados nessa seção visam avaliar o *throughput* e o *packet length* das redes CCN e ETArch executando o caso de uso, visto na Figura 11. Esse caso de uso consiste em uma sequência de troca de mensagens entre os clientes do *Chat* executadas em ordem a fim de compor o tráfego de dados usados para avaliar as arquiteturas comparadas nessa pesquisa.

Diferentemente do IP que tem foco em performance e na característica de conexão *end-to-end* entre a origem e o destino, as redes, ETArch e CCN tem paradigmas diferentes. A abordagem ETArch é centrada no *workflow* pois este é o elemento central capaz de gerenciar e controlar o funcionamento da rede enquanto que a abordagem CCN é considerada orientada a conteúdo, porque é o conteúdo o elemento central de armazenamento e destruição dessa rede.

Para comparar as arquiteturas distintas de distribuição de dados, orientada a *workflow* (ETArch) e orientada a conteúdo (CCN), será criado um cenário de teste comum para as duas abordagens com o intuito de sujeitar cada uma dessas arquiteturas a um mesmo fluxo de dados.

Os pacotes de dados trocados no cenário de teste será analisado usando a ferra-

menta *Wireshark* (versão 1.12.1) de onde será extraído as informações a respeito da taxa de transferência de dados e o tamanho dos pacotes gerados em cada uma das abordagem ([WIRESHARK, 2017](#)).

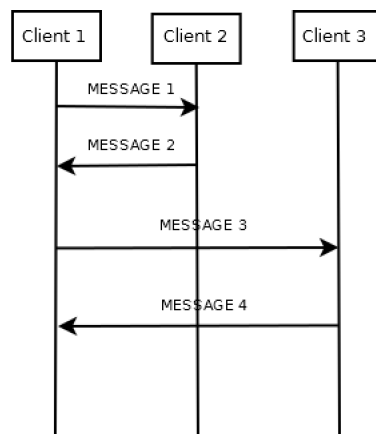


Figura 11 – Exemplo do fluxo de mensagem que será usado para os testes

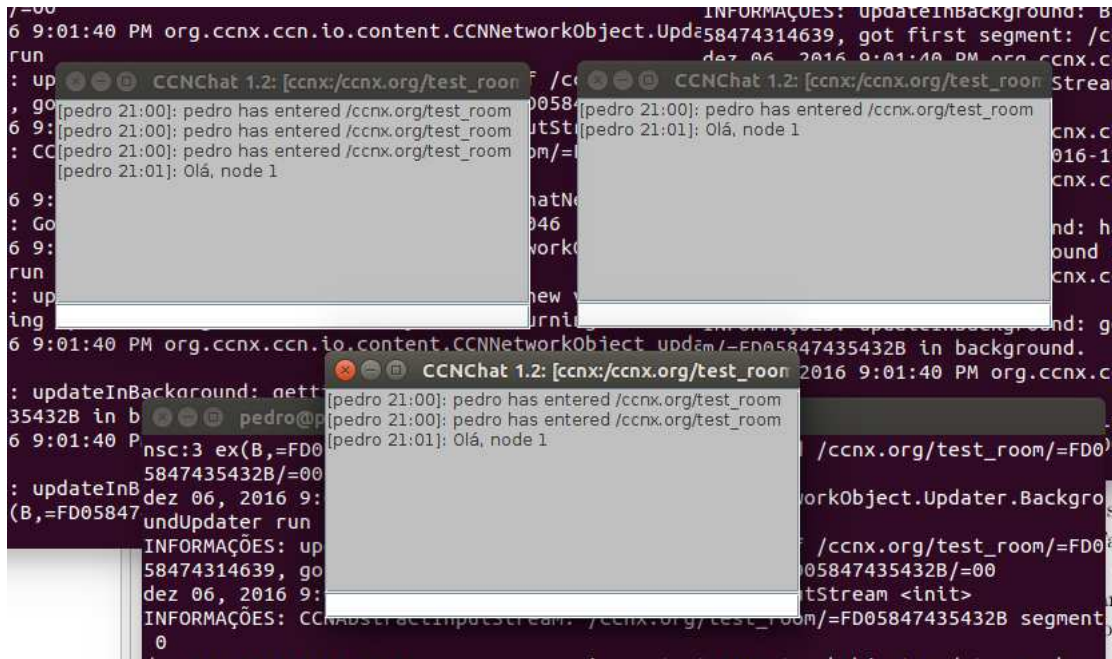
4.4.1 Experimento com a CCN

Já com o ambiente preparado (*Mini-CCNx* instalado) o primeiro passo foi inicializar o *ccnd daemon* com *bin/ccndstart*. Esse é o *script* usado para iniciar o *ccnd daemon* que redireciona as mensagens de *output* para o terminal. Além disso o *ccnd daemon* é responsável por entregar as mensagens *CCNx* para os nós da rede. Depois disso para criar o *Chat* foi necessário apenas definir o nome do *Chat*, por exemplo *ccnchat ccnx:/testroom1* sem a necessidade de informação dos participantes (Figura 12) .

4.4.2 Experimento com a ETArch

Com as máquinas virtuais do *Mininet* e do DTSA do ETArch já configuradas no ambiente de teste foi inicializado o controlador DTSA do ETArch (Figura 14 canto superior esquerdo). Já com o controlador DTSA iniciado a *Mininet* foi usada para criar a topologia da rede definida anteriormente (Figura 10), sendo assim definiu-se os *hosts*, *links*, *switch* e o controlador DTSA (Figura 13). Depois de definida a topologia da rede no *Mininet* cada *host* executou uma aplicação de *Chat* (*script* em python) informando a interface de rede que cada nó pertence, o título da entidade e o título do *workspace*.

Podemos observar no exemplo de execução do *Chat* no ETArch (Figura 14) o fluxo de criação das entidades e *workspaces*. O primeiro nó (h1) foi registrado como a entidade "pedro" e ao tentar se ligar no *workspace w1* ocorre uma falha pois esse *workspace w1* não foi definido. Depois da falha o *workspace w1* é criado e então a entidade "pedro" é ligada

Figura 12 – Exemplo do *Chat CCN*

```

mininet@mininet-vm: ~
*** Adding links:
(h1, s1) (h2, s1) (h3, s1)
*** Configuring hosts
h1 h2 h3
*** Starting controller
*** Starting 1 switches
s1
*** Starting CLI:
mininet> xterm h1
mininet> xterm h2
mininet> xterm h3
mininet> dpctl dump flows
*** s1 -----
ovs-ofctl: unknown command 'dump'; use --help for help
mininet> dpctl dump-flows
*** s1 -----
NXST_FLOW reply (xid=0x4):
 cookie=0x0, duration=79.345s, table=0, n_packets=0, n_bytes=0, idle_age=79, priority=0, dl_src=60:c5:59:0f:72:ee, dl_dst=ff:ff:ff:ff:ff:ff actions=output:1,output:2,output:3
 cookie=0x0, duration=79.35s, table=0, n_packets=1, n_bytes=20, idle_age=69, priority=0, dl_src=60:c5:59:0f:72:ee actions=mod_dl_src:60:c5:59:0f:72:ee, mod_dl_dst:ff:ff:ff:ff:ff:ff, output:1, output:2, output:3
mininet>

```

Figura 13 – Exemplo da criação da topologia de rede ETArch e o fluxo de caminhos definidos no *controller*

ao *workspace w1*. Logo em seguida os nós h2 e h3 registram as entidades "paulo" e "vilela", respectivamente, e se ligam ao *workspace w1*.

4.4.3 Resultados e Discussão

Com a análise dos dados capturados pelo *Wireshark*. Chegamos na Figura 15 que apresenta os pacotes recebidos durante o experimento do *Chat* na arquitetura ETArch. Foi possível observar que a grande parte dos pacotes trocados tinham tamanho entre 40 e 79 bytes. Já no teste feito usando a arquitetura CCN, Figura 16, o mesmo aconteceu,

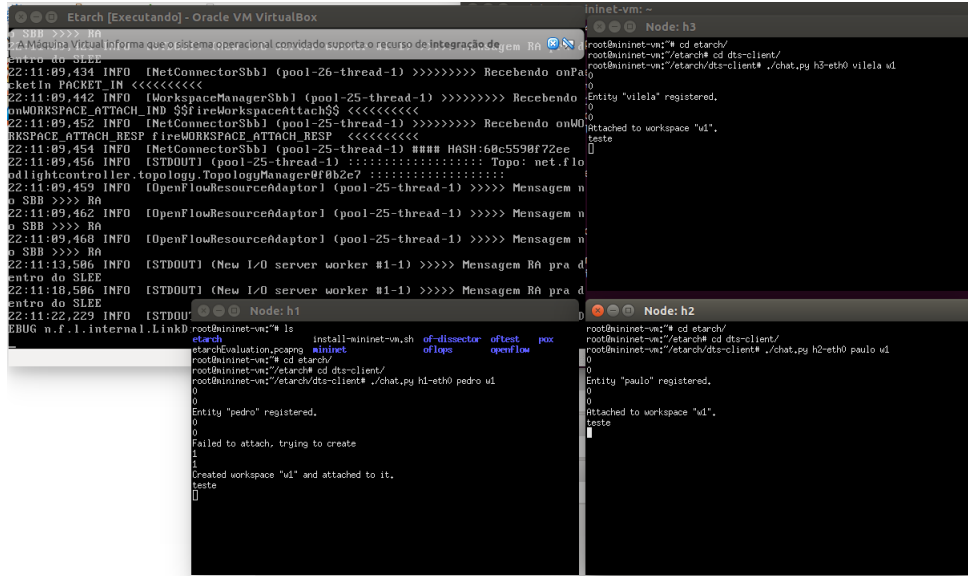


Figura 14 – Exemplo do Chat ETArch

porém com uma quantidade de pacotes trocados entre os *host* menor ao se comparar com arquitetura ETArch, chegando a ser dez vezes menor. Esse fato se deve em parte a grande quantidade de pacotes retransmitidos na arquitetura ETArch observado durante os testes.

Topic / Item	Count	Rate (ms)	Percent
▼ Packet Lengths	2293	0.091917	
0-19	0	0.000000	0.00%
20-39	0	0.000000	0.00%
40-79	846	0.033913	36.89%
80-159	620	0.024853	27.04%
160-319	595	0.023851	25.95%
320-639	32	0.001283	1.40%
640-1279	30	0.001203	1.31%
1280-2559	34	0.001363	1.48%
2560-5119	6	0.000241	0.26%
5120-4294967295	130	0.005211	5.67%

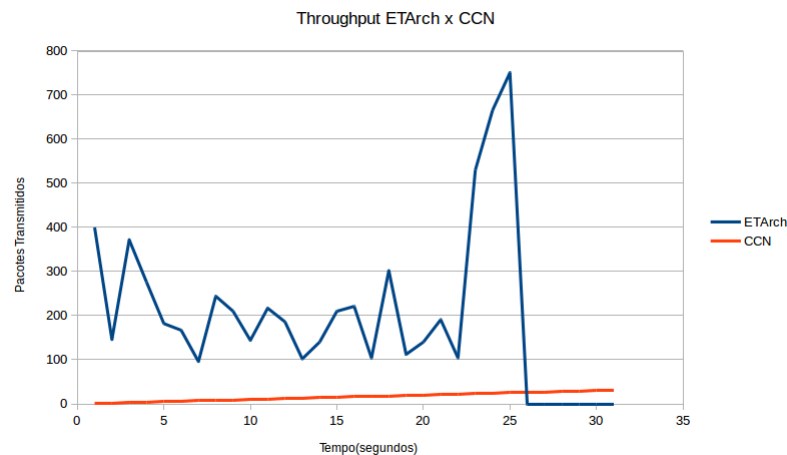
Figura 15 – Tamanho dos pacotes do experimento ETArch

O comportamento das redes CCN e ETArch quanto a quantidade de pacotes transmitidos por segundo (*packets per second* ou PPS) podem ser observados no gráfico 17. É possível notar que a arquitetura ETArch gera uma quantidade maior de pacotes (comparando a coluna *Count* da Figura 15 com a Figura 16), onde foi observado também vários casos de retransmissões de pacotes. Entretanto o tempo para realizar o fluxo de dados da comunicação do *Chat* foi de aproximadamente 26 segundos (como pode se observar no gráfico 17) onde a quantidade de pacotes transmitidos pela ETArch chega a zero. Esse tempo foi menor do que o tempo gasto pela CCN que apesar de gerar uma menor quantidade de pacotes transmitidas no fluxo, após os 26 segundos ainda se tinha pacotes

Topic / Item	Count	Rate (ms)	Percent
▼ Packet Lengths	218	0,007201	
0-19	0	0,000000	0,00%
20-39	0	0,000000	0,00%
40-79	98	0,003237	44,95%
80-159	48	0,001585	22,02%
160-319	40	0,001321	18,35%
320-639	24	0,000793	11,01%
640-1279	8	0,000264	3,67%
1280-2559	0	0,000000	0,00%
2560-5119	0	0,000000	0,00%
5120-4294967295	0	0,000000	0,00%

Figura 16 – Tamanho dos pacotes do experimento CCN

sendo transmitidos exigindo assim um tempo maior para concluir a comunicação (cerca de trinta segundos).

Figura 17 – Gráfico gerado para análise de *throughput* entre ETArch e CCN

Os resultados encontrados no presente estudo sugerem que abordagem ETArch apresentará um maior *throughput* se comparado com a abordagem CCN quando exposta a um tráfego de dados real (tráfego da internet atual) para aplicações de *Chat*. Esse fato pode ser observado pela quantidade de pacotes transmitidos na rede ETArch que chega a ser em média até 10 vezes maior que os pacotes transmitidos da CCN apesar de serem processados de forma mais rápida. Além disso os pacotes transmitidos pela rede ETArch são maiores que o da CCN. Isso indica que a abordagem ETArch quando usada em grande escala poderá apresentar picos de transmissões muito elevados gerando em determinados momentos uma grande quantidade de perdas de pacotes e de retransmissões. A rede CCN por sua vez apresenta uma estabilidade maior durante o fluxo de transmissão de pacotes, apresentando assim uma eficiência maior na utilização da largura de banda se comparado com a ETArch. É possível afirmar, que se pensando em uma rede para substituir a Internet

atual, se tratando de aplicações de *Chat* e de melhor qualidade de serviço(QoS) a CCN está um passo a frente da ETArch.

5 Conclusão

As arquiteturas de rede de Internet do Futuro buscam atender uma série de requisitos não suportados pela Internet atual. A proposta de uma arquitetura *clean slate* para atender os novos requisitos da rede tem sido avaliada pela comunidade de pesquisa envolvida. Dentro desse contexto, o estudo experimental comparativo entre as arquiteturas *clean slate* ETArch e CCN busca apresentar, implementar e testar essas duas abordagens.

Inicialmente foi definida uma metodologia de comparação para as redes ICNs em que a idéia foi criar um ambiente com a implementação das arquiteturas CCN e ETArch para realização de testes. Além disso, foi definido um ambiente de testes apresentando um conjunto de métricas relevantes as duas arquiteturas estudadas, uma aplicação de *Chat* e um cenário de teste. Com o ambiente pronto foi executado um *Chat* em cada uma das arquiteturas aplicando o cenário de teste definido e através da ferramenta *Wireshark* foi feita a captura dos pacotes para serem analisados. Os pacotes de dados capturados foram analisados levando se em conta o *throughput* e *packet length*.

Os resultados mostraram que a abordagem ETArch gerou uma maior quantidade de pacotes, gerando até perdas e retransmissões de muitos desses pacotes, apesar disso o fluxo de dados para o cenário de teste foi executado em um tempo menor que a arquitetura CCN. Esse trabalho é uma pequena contribuição para o início de uma nova geração de redes de Internet do Futuro.

Referências

- AHLGREN, B. et al. A survey of information-centric networking. *IEEE Communications Magazine*, v. 50, n. 7, p. 26–36, July 2012. ISSN 0163-6804. Citado na página 27.
- AHOKAS, K. Software-defined networking. 2014. Disponível em: <<http://kimia.fi/papers/sdn.pdf>>. Citado na página 20.
- AMBIEL, L. M. B.; MAGALHÃES, M. F. Dissertação de Mestrado, *Redes orientadas a conteúdo : uma abordagem no nível de enlace*. 2013. Disponível em: <<http://www.bibliotecadigital.unicamp.br/document/?code=000911854&fd=y>>. Citado 4 vezes nas páginas 6, 13, 22 e 24.
- BRAUN, W.; MENTH, M. Software-Defined Networking Using OpenFlow: Protocols, Applications and Architectural Design Choices. v. 6, n. 2, p. 302–336, 2014. ISSN 1999-5903. Disponível em: <<http://www.mdpi.com/1999-5903/6/2/302/>>. Citado 2 vezes nas páginas 13 e 19.
- CABRAL, C.; ROTHENBERG, C. E.; MAGALHÃES, M. F. Reproducing real NDN experiments using mini-CCNx. In: *Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking*. ACM, 2013. p. 45–46. Disponível em: <<http://dl.acm.org/citation.cfm?id=2491242>>. Citado na página 30.
- CABRAL, C. M. S.; ROTHENBERG, C. R. E. Dissertação de Mestrado, *Mini-CCNx : uma plataforma de prototipagem rápida para redes orientadas à conteúdo*. 2013. Disponível em: <<http://www.bibliotecadigital.unicamp.br/document/?code=000911212>>. Citado 3 vezes nas páginas 6, 22 e 23.
- CASTILLO, J. et al. Evolving Future Internet Clean-Slate Entity Title Architecture with Quality-Oriented Control Plane Extensions. In: . [s.n.], 2014. p. 161–167. ISBN 978-1-61208-360-5. Disponível em: <http://www.thinkmind.org/index.php?view=article&articleid=aict_2014_7_20_10164>. Citado 4 vezes nas páginas 6, 13, 26 e 27.
- DETTI, A. et al. CONET: A Content Centric Inter-networking Architecture. In: *Proceedings of the ACM SIGCOMM Workshop on Information-centric Networking*. ACM, 2011. (ICN '11), p. 50–55. ISBN 978-1-4503-0801-4. Disponível em: <<http://doi.acm.org/10.1145/2018584.2018598>>. Citado na página 13.
- ELLIOTT, C. GENI-global environment for network innovations. In: *LCN*. [s.n.], 2008. p. 8. Disponível em: <<http://euroview2009.com/data/abstracts/Keynote-Elliott-complete.pdf>>. Citado na página 10.
- ETARCH. 2017. Disponível em: <<https://sourceforge.net/projects/etarch/>>. Citado na página 31.
- FARIAS, F. N. et al. Pesquisa experimental para a internet do futuro: Uma proposta utilizando virtualização e o frame-work openflow. 2011. Disponível em: <<http://sbrc2011.facom.ufms.br/files/mc/mc1.pdf>>. Citado 4 vezes nas páginas 10, 11, 13 e 18.

- FERREIRA, C. et al. Towards a Carrier Grade SDN Controller: Integrating OpenFlow With Telecom Services. In: . [s.n.], 2014. p. 70–75. ISBN 978-1-61208-360-5. Disponível em: <http://www.thinkmind.org/index.php?view=article&articleid=aict_2014_3_40_10162>. Citado na página 20.
- FIBRE. *FIBRE Project - Future Internet Testbeds Experimentation Between Brazil and Europe*. 2013. Disponível em: <<http://www.fibre-ict.eu/>>. Citado na página 14.
- FULLER, V.; LI, T. Classless inter-domain routing (CIDR): The internet address assignment and aggregation plan. 2006. Disponível em: <<https://tools.ietf.org/html/rfc4632.txt>>. Citado na página 12.
- GAVRAS, A. et al. Future Internet research and experimentation: the FIRE initiative. v. 37, n. 3, p. 89–92, 2007. Disponível em: <<http://dl.acm.org/citation.cfm?id=1273460>>. Citado 2 vezes nas páginas 10 e 19.
- GUIMARAES, C. et al. IEEE 802.21-enabled Entity Title Architecture for handover optimization. In: *2014 IEEE Wireless Communications and Networking Conference (WCNC)*. [S.l.: s.n.], 2014. p. 2671–2676. Citado na página 19.
- HANDLEY, M. Why the Internet only just works. v. 24, n. 3, p. 119–129, 2006. ISSN 1358-3948, 1573-1995. Disponível em: <<http://link.springer.com/article/10.1007/s10550-006-0084-z>>. Citado na página 10.
- HONG, S.; JANG, M.-W.; LEE, B.-J. CCN networking architecture for mobile applications. In: *2013 IEEE Consumer Communications and Networking Conference (CCNC)*. [S.l.: s.n.], 2013. p. 609–612. Citado 2 vezes nas páginas 13 e 14.
- KANG BYUNGRYEOL SIM, J. K. E. P. W.; LEE, Y. *A Network Monitoring Tool for CCN*. 2012. Disponível em: <<http://docplayer.net/14009658-A-network-monitoring-tool-for-ccn.html>>. Citado 2 vezes nas páginas 6 e 22.
- KOPONEN, T. et al. A Data-oriented (and beyond) Network Architecture. In: *Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. ACM, 2007. (SIGCOMM '07), p. 181–192. ISBN 978-1-59593-713-1. Disponível em: <<http://doi.acm.org/10.1145/1282380.1282402>>. Citado 2 vezes nas páginas 11 e 12.
- LABOVITZ, C. et al. Internet inter-domain traffic. In: . ACM Press, 2010. p. 75. ISBN 9781450302012. Disponível em: <<http://portal.acm.org/citation.cfm?doid=1851182.1851194>>. Citado na página 11.
- LEMA, J. C. *Evolving Future Internet Clean-Slate Entity Title Architecture with Quality-Oriented Control-Plane Extensions*. text, 2014. Disponível em: <https://repositorio.ufrn.br/jspui/bitstream/123456789/18107/1/JoseCL_DISSERT.pdf>. Citado 2 vezes nas páginas 25 e 27.
- LI, S. et al. A comparative study of MobilityFirst and NDN based ICN-IoT architectures. In: *Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine), 2014 10th International Conference on*. IEEE, 2014. p. 158–163. Disponível em: <<http://ieeexplore.ieee.org/abstract/document/6928680/>>. Citado na página 27.

- SILVA, D. A. D. et al. UML-based modeling entity title architecture (ETArch) protocols. 2014. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.677.1420&rep=rep1&type=pdf>>. Citado na página 25.
- SILVA, F. et al. Enabling a Carrier Grade SDN by Using a Top-Down Approach. 2014. Disponível em: <<http://sbrc2014.ufsc.br/anais/files/wpeif/ST2-5.pdf>>. Citado 3 vezes nas páginas 20, 21 e 25.
- SILVA, F. d. O. *Endereçamento por título: uma forma de encaminhamento multicast para a próxima geração de redes de computadores*. text, 2013. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/3/3142/tde-22092014-111409/>>. Citado 8 vezes nas páginas 6, 11, 13, 14, 25, 26, 28 e 31.
- SRISURESH, P.; HOLDREGE, M. IP network address translator (NAT) terminology and considerations. 1999. Disponível em: <https://tools.ietf.org/html/rfc2663_1>. Citado na página 13.
- STERBENZ, J. P. G. et al. Evaluation of network resilience, survivability, and disruption tolerance: analysis, topology generation, simulation, and experimentation: Invited paper. 2011. ISSN 1018-4864, 1572-9451. Disponível em: <<http://link.springer.com/10.1007/s11235-011-9573-6>>. Citado na página 10.
- STUCKMANN, P.; ZIMMERMANN, R. European research on future Internet design. v. 16, n. 5, p. 14–22, 2009. ISSN 1536-1284. Citado na página 10.
- SUñé, M. et al. Design and implementation of the OFELIA FP7 facility: The European OpenFlow testbed. v. 61, p. 132–150, 2014. ISSN 1389-1286. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1389128613004301>>. Citado na página 14.
- VIRTUALBOX. 2017. Disponível em: <<https://www.virtualbox.org/>>. Citado na página 31.
- WIRESHARK. 2017. Disponível em: <<https://www.wireshark.org/>>. Citado na página 32.
- XYLOMENOS, G. et al. A survey of information-centric networking research. *IEEE Communications Surveys Tutorials*, v. 16, n. 2, p. 1024–1049, Second 2014. ISSN 1553-877X. Citado na página 27.
- ZAHARIADIS, T. et al. *Towards a future Internet architecture*. Springer, 2011. Disponível em: <http://link.springer.com/chapter/10.1007/978-3-642-20898-0_1>. Citado na página 10.