

Avaliação Experimental de uma Arquitetura de Rede Recursiva direcionada para Internet do Futuro.

Djalma A. Lima Filho, Carlos Eduardo Dutra

Escola Superior Aberta do Brasil - ESAB

Espírito Santo - ES – Brasil.

dalf@cin.ufpe.br, carlos.dutra@esab.edu.br

Nível: Especialização.

MBA Profissional em Redes de Computadores

Programa de Pós-Graduação da Escola Superior Aberta do Brasil.

Ingresso: 2016.2 Conclusão: 2018.2

Abstract: *The current architecture of the Internet has limitations that interfere with its development. Due to the accentuated growth of new applications demands, the Transmission Control Protocol / Internet Protocol (TCP / IP) model has deficiencies in its structure making it complex. The **Recursive InterNetworking Architecture (RINA)** is an architecture proposal for the Internet of the Future based on interprocess communication that proposes solutions to solve the main limitations in the conventional Internet structure. In this context, this article presents a study the RINA architectures in order to explore the access control in the RINA architecture by using the prototype of the IRATI project through the client server scenario with RINA traffic generation. The results obtained lead to the conclusion that the RINA architecture can be adopted gradually as an alternative to the TCP / IP model to meet the demands of the network applications, besides exploring research areas such as security, multihoming, congestion control and routing.*

Resumo. *A arquitetura atual da Internet apresenta limitações que interferem no seu desenvolvimento. Devido ao crescimento acentuado de novas demandas de aplicações, o modelo TCP/IP (Transmission Control Protocol / Internet Protocol) apresenta deficiências em sua estrutura tornando-o complexo. A Arquitetura Recursiva Interredes (RINA) é uma proposta de arquitetura para a Internet do Futuro baseada na comunicação entre processos que propõe soluções para resolver as principais limitações na estrutura da Internet convencional. Neste contexto, este artigo apresenta um estudo da arquitetura RINA com finalidade de explorar o controle de acesso na arquitetura RINA mediante a utilização do protótipo do projeto IRATI através do cenário cliente servidor com geração de tráfegos RINA. Os resultados obtidos levam à conclusão de que a arquitetura RINA pode ser adotada de maneira gradativa como alternativa frente ao modelo TCP/IP para atender as demandas das aplicações de redes, além de explorar áreas de pesquisas tais como segurança, multihoming, controle de congestionamento e roteamento.*

Palavras-chave: Internet, Arquitetura, IRATI, RINA, TCP/IP.

1. Introdução (Motivação e Justificativa)

A Internet, até o momento, tem evoluído através de “remendos” e essas modificações realizadas na arquitetura têm como finalidade atender novas necessidades das aplicações. Destacamos as limitações do endereçamento e a necessidade de implementação de mecanismos de qualidade de serviço (QoS) sobretudo para as aplicações de tempo real.

Em relação às limitações de endereçamento, o esgotamento de endereços IPv4 foi adiado com a criação do serviço de alocação dinâmica de endereços, através do protocolo *Dynamic Host Configuration Protocol* (DHCP), e a utilização do *Network Address Translation* (NAT). Outro problema relacionado ao endereçamento refere-se à semântica atual que não resolve a separação entre identificação e localização dos pontos de conexão. O IP é utilizado ao mesmo tempo como localizador e identificador e devido a essa sobrecarga, o suporte aos dispositivos móveis se tornou um desafio para a Internet (MOREIRA; FERNANDES, 2009) & (FIARCH, 2010).

A criação do NAT foi uma das soluções intermediárias para solucionar o problema de esgotamento de endereços IPv4. Por outro lado, a criação do IPv6 resolveu de maneira definitiva o problema da escassez do endereço IPv4. No entanto, a sua deficiência persiste quanto à sua semântica.

Em relação à implantação de mecanismos de QoS, um dos pilares básicos da arquitetura da Internet, o princípio do “melhor esforço” (*best effort*) teria que ser abandonado para a priorização de pacotes pertencentes, por exemplo, às aplicações de tempo real. A criação de mecanismos de serviços integrados (IntServ) como também os serviços diferenciados (*DiffServ*) são então exemplos de “remendos” realizados na arquitetura tradicional (FELDMANN, 2007).

Devido aos problemas reportados e às dificuldades para atender novas demandas das aplicações, novos estudos estão sendo feitos com intuito de criar a “Internet do Futuro”. Diante disso, novas arquiteturas de redes estão sendo propostas com finalidade de resolver esses problemas não solucionados. Dentre elas, encontra-se a arquitetura RINA cujo princípio básico é a comunicação entre processos ou IPC (*InterProcess Communication*). Este trabalho tem como objetivo realizar um estudo da arquitetura RINA com o modelo TCP/IP com finalidade de investigar e validar de forma experimental as características da arquitetura RINA utilizando protótipo do projeto IRATI¹ para analisar o comportamento e as interações dos seus componentes.

O restante deste artigo está organizado da seguinte maneira. Na Seção 2 é abordada a arquitetura RINA apresentando a motivação que levou ao seu desenvolvimento, assim como, suas características e uma comparação com a pilha de

¹ Projeto IRATI - Disponível em: <http://irati.eu/>

protocolos da Internet que representa a arquitetura convencional bem como o estado da arte indicando os trabalhos relacionados. Por fim, na Seção 3 é apresentada a metodologia de avaliação adotada apresentando o estudo de caso como também a avaliação dos resultados.

2. A arquitetura RINA & Estado da Arte

Os princípios da arquitetura RINA foram apresentados em (DAY, 2008). RINA defende o uso de uma única camada chamada DIF (*Distributed IPC Facility*) que fornece serviço de comunicação entre processos, IPC, de maneira distribuída, seja em uma rede de pequena distância ou uma rede de longa distância. As DIFs podem ser empilhadas com as mesmas funcionalidades e os mesmos mecanismos de forma recursiva cobrindo diferentes escopos. A Figura 1 apresenta uma diferença entre as camadas da arquitetura RINA em relação arquitetura TCP/IP. É possível observar que o empilhamento das camadas na estrutura da Internet atual utilizam vários protocolos de comunicação e cada camada apresenta uma função específica seja de transporte, roteamento e correção de erros. Na RINA todas as camadas são caracterizadas para prover a comunicação dos processos IPCs o que permite uma flexibilidade em sua estrutura para as aplicações de rede e além do mais essas camadas utilizam os mesmos mecanismos com políticas diferenciadas para tratar no congestionamento da rede, roteamento e dentre outras áreas.

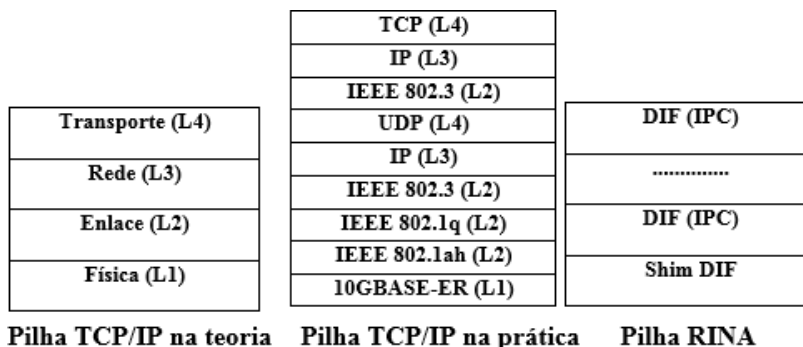


Figura 1: Diferença da estrutura de camadas das arquiteturas RINA e TCP/IP.

Segundo Day (2008) cada processo IPC que precisa estabelecer um fluxo de comunicação, tem que fazê-lo através de um outro processo subjacente através de uma DIF. Cada processo consiste em três conjuntos distintos de tarefas que lidam com aspectos diferentes. Estes conjuntos de tarefas são armazenadas em uma base de dados conhecida como RIB (*Resource Information Base*). A transferência de dados IPC utiliza o módulo RMT (*Relaying and Multiplexing Task*) que gerencia o encaminhamento e a

multiplexação de todas as PDUs (*Protocol Data Units*). O controle de transferência IPC implementa o controle de erro e de fluxo, além de monitorar os parâmetros de transferência de dados por fluxo que é realizado pelo protocolo EFCP (*Error and Flow Control Protocol*). A gerência IPC implementa o protocolo de aplicação distribuído comum para consultar e atualizar a RIB. Os protocolos na RINA utilizam mecanismos cujos comportamentos são definidos por políticas. Por exemplo, um mecanismo utilizado pelo protocolo EFCP é o controle de fluxos e congestionamento. Por outro lado, a alocação desses fluxos é uma política implementada pelo mesmo protocolo. Diante disso, a arquitetura se torna menos complexa em relação ao modelo convencional por utilizar um conjunto básico de mecanismos para diversos escopos (DAY, 2008).

Alguns trabalhos tais como Vrijders et al. (2014), Luiz; Monteiro (2016), Bergesio (2013), e Wang; Esposito; Matta (2013) foram realizados para explorar as funcionalidades da arquitetura RINA utilizando redes de experimentação (*testbeds*) e *frameworks* ProtoRINA² como também o protótipo IRATI. Nos trabalhos de Bergesio (2013), Wang; Esposito; Matta (2013) e Wang; Akhtar; Matta (2014), especificamente, os autores abordaram toda implementação do protótipo IRATI³ fazendo uma avaliação da arquitetura RINA através de *testbeds* analisando aspectos de roteamento e fazendo avaliação de desempenho sobre concorrência de fluxos de dados RINA e IP.

3. Estudo de Caso & Avaliação dos Resultados

Para investigação dos componentes da arquitetura RINA, o trabalho realizado permitiu a instalação e configuração do protótipo *stack IRATI* disponível no endereço <https://github.com/IRATI/stack> aplicando a medição como técnica de avaliação para coletas dos tempos de ida-e-volta dos fluxos de dados (RTT – *Round Trip Time*). Portanto, duas máquinas virtuais (VM) com instalação *GNU/Debian Jessie* 8 e *kernel* 4.1.16 foram preparadas e configuradas para o ambiente RINA. As VM foram configuradas utilizando software de virtualização *virtual box*⁴. Para gerar tráfegos RINA foi utilizada uma aplicação denominada *rina-echo-time* para coleta dos RTTs. Assim, o ambiente configurado e as ferramentas de geração de tráfego estão representados na Figura 2.

² ProtoRINA: Disponível em: <https://github.com/ProtoRINA/users/wiki>.

³ Protótipo IRATI – Disponível em: <https://github.com/IRATI/stack>.

⁴ Virtual Box. Disponível em: <https://www.virtualbox.org/>.



Figura 2: Estudo de Caso - Topologia de Avaliação.

Para geração de tráfego entre as aplicações cliente-servidor é necessário que o processo de aplicação que queira se comunicar com outro processo solicite essa comunicação. A validação dessa comunicação é realizada através do registro na camada DIF que será autorizada ou não pelo processo IPC residente na camada. Depois que autorização é concedida a aplicação pode trocar informações. Portanto, a Figura 3 apresenta o registro realizado pelo processo IPC “test1.IRATI” registro esse solicitado pelo processo IPC “test2.IRATI”, enquanto que a Figura 4 apresenta a comunicação dos processos IPCs após registro ser concedido com sucesso.

```

djalma@ArchRinaOne: ~
djalma@ArchRinaOne:~$ telnet localhost 32766
Trying ::1...
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
IPCM >>> list-ipcps
Current IPC processes (id | name | type | state | Registered applications | Port-ids of flows provided)
 1 | test-eth-vlan:1: | shim-eth-vlan | ASSIGNED TO DIF 100 | test1.IRATI-1-- | -
 2 | test1.IRATI:1:: | normal-ipc | ASSIGNED TO DIF normal.DIF | - | -
IPCM >>>
enroll-to-dif 2 normal.DIF 100 test2.IRATI 1IPCM >>>
DIF enrollment succesfully completed

```

Figura 3: Screenshot Registro autorizado com sucesso para aplicação “test2.IRATI”.

```

djalma@ArchRinaTwo: ~
root@ArchRinaTwo:/home/djalma# rina-echo-time -c 15 -s 512
18480(1474076721)#librina.logs (DBG): New log level: INFO
18480(1474076721)#librina.nl-manager (INFO): Netlink socket connected to local port 18480
Flow allocation time = 21.035 ms
SDU size = 512, seq = 0, RTT = 2.2042 ms
SDU size = 512, seq = 1, RTT = 6.872 ms
SDU size = 512, seq = 2, RTT = 6.4962 ms
SDU size = 512, seq = 3, RTT = 12.506 ms
SDU size = 512, seq = 4, RTT = 15.425 ms
SDU size = 512, seq = 5, RTT = 8.8212 ms
SDU size = 512, seq = 6, RTT = 10.495 ms
SDU size = 512, seq = 7, RTT = 9.8934 ms
SDU size = 512, seq = 8, RTT = 10.965 ms
SDU size = 512, seq = 9, RTT = 10.687 ms
SDU size = 512, seq = 10, RTT = 9.3322 ms
SDU size = 512, seq = 11, RTT = 14.778 ms
SDU size = 512, seq = 12, RTT = 14.747 ms
SDU size = 512, seq = 13, RTT = 16.579 ms
SDU size = 512, seq = 14, RTT = 14.986 ms
SDUs sent: 15; SDUs received: 15; 0% SDU loss
Minimum RTT: 2.2042 ms; Maximum RTT: 16.579 ms; Average RTT:10.986 ms; Standard deviation: 3.9729 ms
SDUs sent: 0; SDUs received: 0; 0% SDU loss
Minimum RTT: 2.1475e+09 ms; Maximum RTT: 0 ms; Average RTT:0 ms; Standard deviation: -0 ms
root@ArchRinaTwo:/home/djalma#

```

Figura 4: Screenshot da Comunicação entre os processos “test1.IRATI” e “test2.IRATI”.

Em comparação com o modelo TCP/IP, caso fosse necessário trocar informações com as aplicações seria imprescindível um protocolo de segurança para que essas

informações sejam trafegadas na camada de maneira segura. Na RINA, isso não acontece, pois após o registro realizado na DIF o fluxo de dados são visíveis apenas as aplicações na mesma camada. Portanto, esse trabalho permitiu explorar as funcionalidades na RINA bem como validar o controle de acesso dos processos IPCs. Como trabalhos futuros é possível explorar a arquitetura fazendo uma avaliação de desempenho da transferência de dados no cenário de redes de longa distância para analisar comportamento da arquitetura.

Referências

- Bergesio, L. Prototipo de RINA sobre Ethernet. Universitat Oberta de Catalunya - UOC, 2013.
- Day, J. Patterns in Network Architecture: A Return to Fundamentals. 1. ed. Westford, Massachusetts.: Pearson Education, 2008.
- Feldmann, A. Internet Clean-Slate Design: What and Why? *Computer Communication Review*, v. 37, n. 3, p. 59–64, 2007.
- Fiarch, E. Fundamental Limitations of Current Internet and the path to Future Internet 1. p. 1–11, 2010.
- Luiz, C.; Monteiro, J. A. S. Investigação do Protótipo ProtoRINA utilizando o Ambiente FIBRE. *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, p. 1–4, 2016.
- Moreira, M.; Fernandes, N. Internet do futuro: Um novo horizonte. *Minicursos do Simpósio ...*, p. 1–59, 2009.
- Vrijders, S. et al. Experimental evaluation of a Recursive InterNetwork Architecture prototype. *2014 IEEE Global Communications Conference, GLOBECOM 2014*, p. 2017–2022, 2014.
- Wang, Y.; Akhtar, N.; Matta, I. Programming routing policies for video traffic. *Proceedings - International Conference on Network Protocols, ICNP*, p. 504–510, 2014.
- Wang, Y.; Esposito, F.; Matta, I. Demonstrating RINA using the GENI testbed. *Proceedings - 2013 2nd GENI Research and Educational Experiment Workshop, GREE 2013*, p. 93–96, 2013.