12º Congresso de Inovação, Ciência e Tecnologia do IFSP - 2021

Análise Passiva de redes de comunicação industrial PROFINET

Marcio Rafael Buzoli[1], Andre Luis Dias[2], Guilherme Serpa Sestito[3], Afonso Celso Turcato[4]

[1] Graduando em Engenharia Elétrica do IFSP Câmpus Sertãozinho, m.buzoli@aluno.ifsp.edu.br
[2] Professor e pesquisador do IFSP Câmpus Sertãozinho, andre.dias@ifsp.edu.br
[3] Pós - Doutorando Universidade de São Paulo e Professor do IFSP Câmpus Sorocaba, guilherme.sestito@ifsp.edu.br
[4] Professor e pesquisador do IFSP Câmpus Sertãozinho, afonso.turcato@ifsp.edu.br
Área de conhecimento (Tabela CNPq): 1.03.04.03-7 Software Básico

RESUMO:

Esse trabalho tem por objetivo desenvolver uma ferramenta para a análise passiva de redes de comunicação industrial PROFINET. Esse protocolo é mantido pela associação de empresas de automação industrial e universidades, denominada PROFIBUS & PROFINET International (PI). O protocolo PROFINET é amplamente reconhecido como um dos mais utilizados atualmente em diversas aplicações e segmentos de mercado. A utilização de redes Ethernet industriais para automação traz muitas vantagens além da padronização. Principalmente em sistemas onde a variável tempo é um fator crítico, a tecnologia PROFINET oferece um modelo de dispositivos modulares em uma troca cíclica de informações entre seus dispositivos. Através da correta interpretação dos dados de tráfego de rede, é possível gerar um relatório com uma análise que proporciona aos usuários informações valiosas em diferentes cenários de utilização dessa tecnologia, dessa forma ajudando a reduzir os prejuízos financeiros provenientes de paradas não programadas na produção.

PALAVRAS-CHAVE: PROFINET; Real Time Ethernet; Python; PCAP; Captura de tráfego.

PROFINET Network Passive Analysis

ABSTRACT:

This work aims to develop a tool for the passive analysis of PROFINET industrial communication networks. This protocol is maintained by the association of industrial automation companies and universities, called PROFIBUS & PROFINET International (PI). The PROFINET protocol is widely recognized as one of the most used in various applications and market segments. The use of industrial Ethernet networks for automation provides many advantages beyond standardization. Mainly in systems which require critical time constraints, PROFINET technology offers a modular device model in a cyclical data exchange between its devices. Through the correct interpretation of network data traffic, it is possible to generate a report with an analysis that provides users with valuable information in different scenarios where this technology is employed, thus helping to reduce the financial losses arising from unscheduled production stops.

KEYWORDS: PROFINET; Real Time Ethernet; Python; PCAP; Traffic sniffing.

INTRODUÇÃO

Este estudo tem por objetivo o desenvolvimento de uma ferramenta para análise passiva de protocolos industriais baseado em Real-time Ethernet mais especificamente a tecnologia PROFINET. A análise será realizada por meio de um algoritmo desenvolvido em linguagem Python, e sua validação por meio de coletas e análises de dados provenientes de redes reais coletadas plantas dos mais diversos setores, como automotivo e sucro alcooleiro.. A ferramenta será acessível a todos que tenham interesses exploratórios ou práticos da tecnologia estudada, permitindo uma análise passiva de redes PROFINET

a partir de coleta de tráfego, preferencialmente no link entre o controlador e os demais dispositivos da rede. Para essa coleta, recomenda-se o uso do software *open source* denominado Wireshark.
O algoritmo do estudo realizará toda sua análise utilizando as informações contidas nos arquivos salvos pelo software Wireshark (formato .pcap) para transformá-los em metadados organizados e de fácil interpretação do usuário, ao término dessa análise será gerado um arquivo PDF (Portable Document Format) para download onde serão abordados os seguintes indicadores de rede: Lista de dispositivos na rede identificados pelo *MAC Address*; Tempo de ciclo médio e desvio padrão de cada dispositivo; Jitter na comunicação de cada dispositivo; Quantidade média de pacotes por segundo; Quantidade de Pacotes recebidos/enviados por cada dispositivo; Ocupação de banda total da rede (%); Ocupação de banda por pacotes Real Time (%); Ocupação de banda por pacotes non Real Time (%); Pacotes perdidos; Identificação de alarmes de alta e baixa prioridade.

MATERIAL E MÉTODOS

Cada linguagem possui facilidades e propósitos distintos, o Python é uma ferramenta *open-source* que possui características interessantes aliada a uma expressiva comunidade ativa, trata-se de uma linguagem simples e poderosa amplamente utilizada em *data science, machine learning* e desenvolvimento Web.

A escolha do Python da necessidade em publicar uma ferramenta web para analise passiva e também por possuir um crescente número de bibliotecas de código aberto(módulos) disponibilizadas online gratuitamente no repositório Pypi (https://pypi.org/). Durante esse estudo foram identificadas algumas bibliotecas as quais utilizamos na confecção do algoritmo, seguem abaixo as principais:

• **Scapy** – Versão 2.4.5(https://scapy.net/)
manipulação de arquivos *.pcap (*.cap, *.pcapng, entre outros).
• **Matplotlib** – Versão 3.4.2 (https://matplotlib.org/): O Matplotlib é uma biblioteca para geração de gráficos 2D a partir de arrays em linguagem Python.
• **Pyfpdf** – versão 1.7.2 (https://code.google.com/p/pyfpdf/): Biblioteca para geração de documentos PDF em Python.

O desenvolvimento do código foi iniciado por meio da extração dos dados contidos no arquivo PCAP. Analisando as informações provenientes de uma rede PROFINET é possível reconhecer o dispositivo que enviou a informação (“src”), o dispositivo que recebeu essa informação (“dst”), tipo do protocolo (“type”) e a informação em linguagem de máquina do dispositivo em questão (Raw):

No repositório da biblioteca Scapy no GitHub é disponibilizado a instanciação de objetos presentes em arquivos PCAP, instanciar um objeto é a ação de criar um exemplar de uma classe para trabalhar com ele. Após a devida configuração para nossos objetivos, o comando “*.show2()*” nos trouxe informações adicionais:

```
print(packets[0].show2())
```

```
###[ Ethernet ]###
  dst       = 00:0e:8c:bd:32:de
  src       = 00:0e:8c:f7:53:b6
  type      = 0x8892
###[ ProfinetIO ]###
     frameID   = RT_CLASS_1 (8010)
###[ PROFINET Real-Time ]###
        \data      \
         |###[ PROFINET IO Real Time Cyclic Default Raw Data ]###
         |  data      = '\x80\x80\x80\x80\x80\x00\x00\x80\x00\x00\x00\x00\x00\x00\x
00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x0c\x085\x00\x06\x03\xe6/\xc8>\xa7\x00\x01\x17@\x00N\x96\x02\xa
e'
        padding   = ''
        cycleCounter= 2561
        dataStatus=
        transferStatus= 0
```

Também através da biblioteca Scapy foi possível acessar o epoch time de cada captura:

```
print(packets[0].time)
```

```
Decimal('1588632510.644733')
```

Todo o relatório é fundamentado nas informações apresentadas nessa etapa. Os primeiros passos do código visam filtrar as informações irrelevantes para o estudo, para isso foi utilizado uma estrutura de repetição aninhada e algumas condicionais que retiram da lista conexões com protocolos diferentes do PROFINET (identificado pelo Ethertype *0x8892*):

RESULTADOS E DISCUSSÃO

São diversos os aspectos que tornam o PROFINET uma poderosa ferramenta, toda sua capacidade deriva de inúmeras nuances em sua instalação, configuração e manutenção, por esse motivo a coleta de dados em um ambiente controlado são de suma importância para possíveis ajustes finais ao algoritmo desenvolvido. Todos indicadores foram alcançados utilizando dados de arquivos simulados e de coletas em ambiente controlado.

A análise é separada em quatro etapas:

1. Análise geral da rede.
2. Lista dos indicadores em cada conexão, considerando um dispositivo fonte e o dispositivo de destino (src → dst), cada um identificado como um "*communication ID"*.
3. Gráfico dos pacotes enviados em função do tempo (ordem cronológica).
4. Informações individuais de cada conexão (src → dst), indicando pacotes perdidos e alarmes diversos.

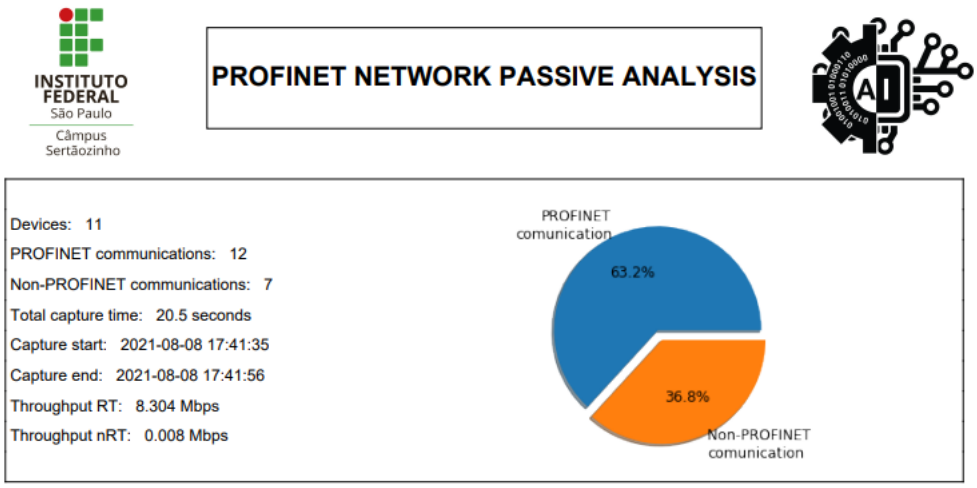

Figura 1. Análise geral da rede

Posteriormente os dados são dispostos de acordo com seu "*communication ID*", esse ID é definido de acordo com a ordem de leitura do arquivo enquanto é criado a Lista "CL", nesse momento são listados apenas comunicações PROFINET. Este parâmetro visa informar ao usuário os links de comunicação estabelecidos entre os dispositivos presentes na rede. Vale lembrar que comunicações não PROFINET recebem ID, porém não serão abordadas nessa etapa do estudo. Complementarmente, são informados ao usuário do software desenvolvido, os indicadores de desempenho de redes RTE, como: tempo de ciclo denominado de Time of Data Exchange (TDE), Jitter, Throughput. Além da quantidade de alarmes gerados.

PROFINET communications analysis							
Comunication ID	**Source**	**Destination**	**TDE SET [ms]**	**TDE AVG [ms]**	**Jitter [µs]**	**Throughput**	**Alarm(s)**
1	00:0e:8c:bd:32:db	00:0e:8c:f7:53:b6	8	7.99	14.83	0.080 Mbps	0
2	00:0e:8c:bd:32:de	00:0e:8c:f7:53:b6	8	7.99	14.45	0.080 Mbps	0
3	00:0e:8c:e1:9c:9d	00:0e:8c:f7:53:b6	8	8.00	14.52	0.158 Mbps	0
5	00:0e:8c:ea:39:5c	00:0e:8c:f7:53:b6	8	7.99	37.93	0.080 Mbps	0
6	00:0e:8c:f6:96:96	00:0e:8c:f7:53:b6	8	7.99	15.85	0.080 Mbps	0
7	00:0e:8c:f7:53:b6	00:0e:8c:bd:32:db	8	7.99	28.62	0.080 Mbps	0
8	00:0e:8c:f7:53:b6	00:0e:8c:bd:32:de	8	7.99	27.62	0.080 Mbps	0
9	00:0e:8c:f7:53:b6	00:0e:8c:e1:9c:9d	8	7.99	23.76	0.080 Mbps	0
10	00:0e:8c:f7:53:b6	00:0e:8c:ea:39:5c	8	7.99	114.71	0.080 Mbps	0
11	00:0e:8c:f7:53:b6	00:0e:8c:f6:96:96	8	7.99	111.53	0.080 Mbps	0
12	00:0e:8c:f7:53:b6	00:1f:f8:04:06:f2	8	7.99	102.12	0.080 Mbps	0
14	00:1f:f8:04:06:f2	00:0e:8c:f7:53:b6	8	7.99	14.81	0.080 Mbps	0

Figura 2. Lista dos indicadores em cada conexão src → dst

Através de um gráfico de dispersão onde cada círculo representa um pacote de dados, o gráfico da figura 3 foi elaborado para tornar visual todos os pacotes de dados entre os 2 dispositivos de cada "communication ID". O eixo x representa o horário exato da captura do pacote, já o eixo y é referente ao tempo de intervalo entre o pacote representado e o pacote anterior. Simbolizado pela linha preta, o valor de 8ms refere-se ao valor parametrizado em software para aquela comunicação, dessa forma, quanto mais próximo da linha central, mais próximo o tempo de intervalo entre os pacotes estará do ideal. Quanto mais afastado o pacote estiver dos 8ms, mais intensa será a cor representada, se for abaixo de 8ms a intensidade é voltada para tons de azul, mas caso seja acima de 8ms, a cor irá variar em tons de vermelho, para essas variações de tempo, dá-se o nome de jitter, um importante indicador de desempenho de redes Real Time Ethernet. Na Figura 2 ele era representado numericamente, na Figura 3 pode-se visualizar seu comportamento graficamente.

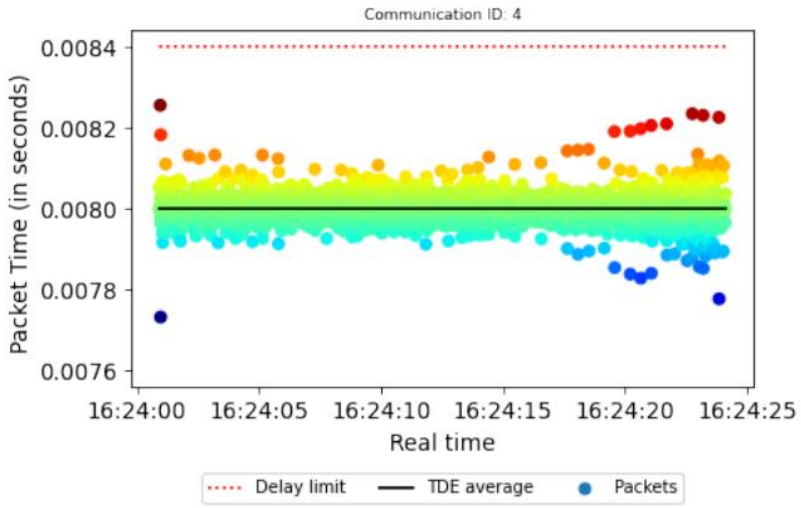

Figura 3. Gráfico dos pacotes em ordem cronológica.

Por fim, a implementação do quadro de descrição dos alarmes faz-se necessária quando erros acontecem e o usuário final precisa entender sua origem. O algoritmo desenvolvido conta com a descrição dos alarmes de alta e baixa prioridades que ocorreram durante a captura dos pacotes, através dessa funcionalidade é possível realizar manutenções preditivas uma vez que o gerenciamento de informações dos alarmes é uma poderosa ferramenta para o entendimento da dinâmica de falhas daquela rede em específico. Pode-se ver, como exemplo, a Figura 4 informando sobre o alarme "Power supply fault" ou em uma tradução literal, falha na fonte de alimentação.

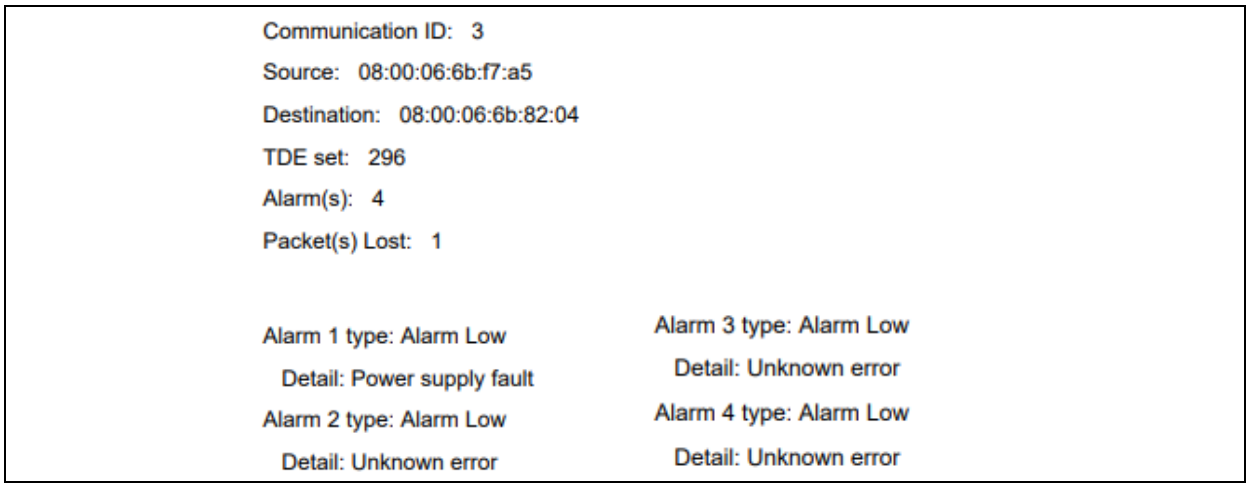

Figura 4. Informações individuais de cada conexão src → dst, indicando pacotes perdidos e alarmes).

CONCLUSÕES

O uso de protocolos industriais baseados em *Real Time Ethernet* (RTE) segue em crescimento impulsionados pelas novas demandas de implementações para ferramentas da Indústria 4.0, Big Data e Internet das Coisas, e entre outros protocolos, o protocolo PROFINET é amplamente reconhecido como um dos mais utilizados por atuar em diversas aplicações na área de automação industrial, e subsequentemente aumentar a demanda por profissionais capacitados para acompanhar tal crescimento.

Os dados obtidos através da análise têm caráter exploratório para indústrias e entusiastas da comunidade cientifica que utilizam a tecnologia PROFINET em sua rotina, o relatório ao final da análise constitui um diagnóstico das principais funcionalidades com base em seus dados e serve como material de apoio nas investigações da rede analisada.

Como trabalho futuro, busca-se a utilização da ferramenta desenvolvimento em diferentes cenários, como número de dispositivos, topologias, e mesmo equipamentos de coleta do tráfego da rede. Assim, verificando a influência destas variáveis dos cenários no desempenho destas redes.

AGRADECIMENTOS

Os autores agradecem o suporte acadêmico e a estrutura para pesquisa do Instituto Federal de São Paulo Campus Sertãozinho e de Sorocaba.

REFERÊNCIAS

DIAS, A.L., SESTITO, G.S. & BRANDÃO, D. Performance Analysis of Profibus DP e Profinet in a Motion Control Application. J Control Autom Electr Syst 28, 86-93 (2017). Disponível em: <https://doi-org.ez338.periodicos.capes.gov.br/10.1007/s40313-016-0278-7>.

J. D. Hunter, "Matplotlib: A 2D Graphics Environment", Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007.

SCAPY - © Direitos Autorais 2008-2021 Philippe Biondi e a comunidade Scapy Disponível em: <https://scapy.readthedocs.io/en/latest/backmatter.html>.

A. B. Lugli, E. R. Neto, J. P. C. Henriques, M. T. d. C. Silva, N. D. Pereira and T. C. Pimenta, "A Database Proposal for an Application Involving Industrial Networks for Industry 4.0 Concepts," 2020 27th International Conference on Mixed Design of Integrated Circuits and System (MIXDES), 2020, pp. 239-244, doi: 10.23919/MIXDES49814.2020.9155979.

PROFIBUS & PROFINET International (PI), "PROFINET Installation Guidelines: date 21/11/2019 filename PROFINET_Planungsrichtlinie_8061_V138_Sep19.pdf" disponível em: <https://www.profibus.com/download/profinet-installation-guidelines/>.