



Unidade de Pós-Graduação, Extensão e Pesquisa



Heimdall: Detecção de Artefatos Maliciosos Utilizando Machine Learning em Ambientes Internet of Things Habilitados por Redes Definidas por Software

Cristian Henrique Martins de Souza

Orientador: Prof. Dr. Carlos Hideo Arima

Agenda

1. Introdução
2. Motivação
3. Fundamentação teórica
4. Proposta
5. Validação e análise
6. Conclusão

Introdução

Introdução

- A quarta revolução industrial é caracterizada pela incorporação de tecnologias emergentes para automação de tarefas.
 - Internet of Things (IoT), *big data*, inteligência artificial, entre outras.
- Essas tecnologias trazem ganhos significativos para os sistemas produtivos.
 - Aumento da eficiência no uso de recursos e no desenvolvimento de produtos em larga escala.
- Com o advento da indústria 4.0, foi necessário o desenvolvimento de normas e padrões para orientar as etapas de projetos e protótipos.
 - Reference Architecture Model for Industry 4.0 – RAMI 4.0.
- Tendo em vista que dispositivos IoT são um dos pilares da Indústria 4.0, a proteção deles é essencial para a segurança da informação e dos seus utilizadores.
- Entretanto, essa é uma tarefa complexa, visto que dispositivos IoT possuem um baixo poder de processamento e comumente não recebem atualizações de segurança por parte dos fornecedores.
- Ademais, a heterogeneidade de protocolos demanda o uso de abordagens adaptativas e com visão holística da infraestrutura para mitigação efetiva de ameaças modernas.

Introdução

- Malwares continuam sendo um dos principais desafios à segurança de sistemas computacionais.
- O advento do paradigma IoT foi acompanhado pelo aumento do número de programas maliciosos projetados para as arquiteturas ARM e MIPS.
 - Um exemplo disso é a utilização da botnet Mirai.
- Essas ameaças são responsáveis por diversos danos, como o comprometimento da integridade dos dados, roubo de informações confidenciais e prejuízos financeiros a usuários e corporações.
- Ferramentas de detecção devem ser capazes de identificar e conter ameaças desconhecidas.
- Nesse contexto, o uso de machine learning tem se mostrado efetivo na detecção genérica de artefatos maliciosos em nível de rede.

Introdução

- Para isso, um modelo é treinado com base em uma grande quantidade de dados a respeito de ameaças conhecidas.
- Uma vez treinado, ele é capaz de realizar previsões assertivas ao lidar com novas informações.
- Isso permite a detecção e classificação efetiva das ameaças com base na análise de tráfego, minimizando o risco de comprometimentos.
- Aliado a isso, paradigmas como o das Redes Definidas por Software têm viabilizado o desenvolvimento de soluções adaptativas e capazes de identificar ameaças na rede de forma holística.
- O potencial das ferramentas de detecção pode ser elevado ainda mais por meio de switches programáveis (e.g., via a linguagem P4).

Introdução

Principais alterações:

1. Inclusão de termos sobre sistemas produtivos nas seções especificadas pelos examinadores;
2. Inclusão de referências mais atuais na subseção de *machine learning*;
3. Melhorias na seção de metodologia científica;
4. Correções nas legendas das imagens;
5. Correções gerais no texto e estrutura do trabalho.

Motivação

Motivação

- A partir de uma revisão da literatura, foi possível constatar que a maior parte das soluções para detecção das ameaças de malware em ambientes IoT habilitados por SDN propostas pela academia estão diretamente atreladas ao controlador SDN.
- Dessa forma, caso ele seja alvo direto de ataques, especialmente os de negação de serviço, a segurança e operação da rede podem ser comprometidas.
- Logo, é necessário o desenvolvimento de soluções mais resilientes a ataques direcionados, visando manter a estabilidade e segurança da rede.
- Inspirado pelo poder e flexibilidade dos switches programáveis, este trabalho propõe o Heimdall, uma ferramenta para detecção de artefatos maliciosos em ambientes IoT habilitados pela tecnologia SDN.

Motivação

- Para isso, é adotada uma abordagem híbrida, composta pela detecção de assinaturas maliciosas e pela classificação do tráfego por meio de machine learning.
- Diferentemente das abordagens tradicionais, a solução é acoplada diretamente aos switches da rede, o que reduz a ocorrência de pontos únicos de falha e otimiza a utilização dos recursos presentes na infraestrutura.
- A ferramenta é avaliada em um ambiente controlado e emulado por meio da ferramenta Mininet, juntamente com o auxílio de switches virtualizados.
- São utilizados exemplares de malwares reais para validar a acurácia da solução.

Motivação

Com base no exposto, este trabalho é fundamentado nas seguintes proposições:

- **Proposição 1:** a detecção de artefatos maliciosos em ambientes IoT habilitados por SDN pode ser aprimorada com o auxílio de uma abordagem baseada em assinaturas e machine learning acoplada diretamente em comutadores programáveis.
- **Proposição 2:** soluções para detecção e mitigação de programas maliciosos podem ser desenvolvidas sem a necessidade de um controlador SDN, eliminando pontos únicos de falha.

Motivação

- **As principais contribuições deste trabalho são:**
 1. Avaliar diferentes algoritmos de machine learning em um dataset robusto, visando identificar o modelo mais eficaz na classificação de programas maliciosos;
 2. Propor uma arquitetura híbrida, genérica e resiliente para detecção de malwares baseada em assinaturas maliciosas e machine learning;
 3. Desenvolver uma solução como prova de conceito para detecção de programas maliciosos em ambientes IoT habilitados por SDN que é acoplada diretamente aos switches da rede;
 4. Avaliar a abordagem híbrida proposta a partir de exemplares reais de artefatos maliciosos.

Fundamentação teórica

Fundamentação teórica

- Trabalhos relacionados:

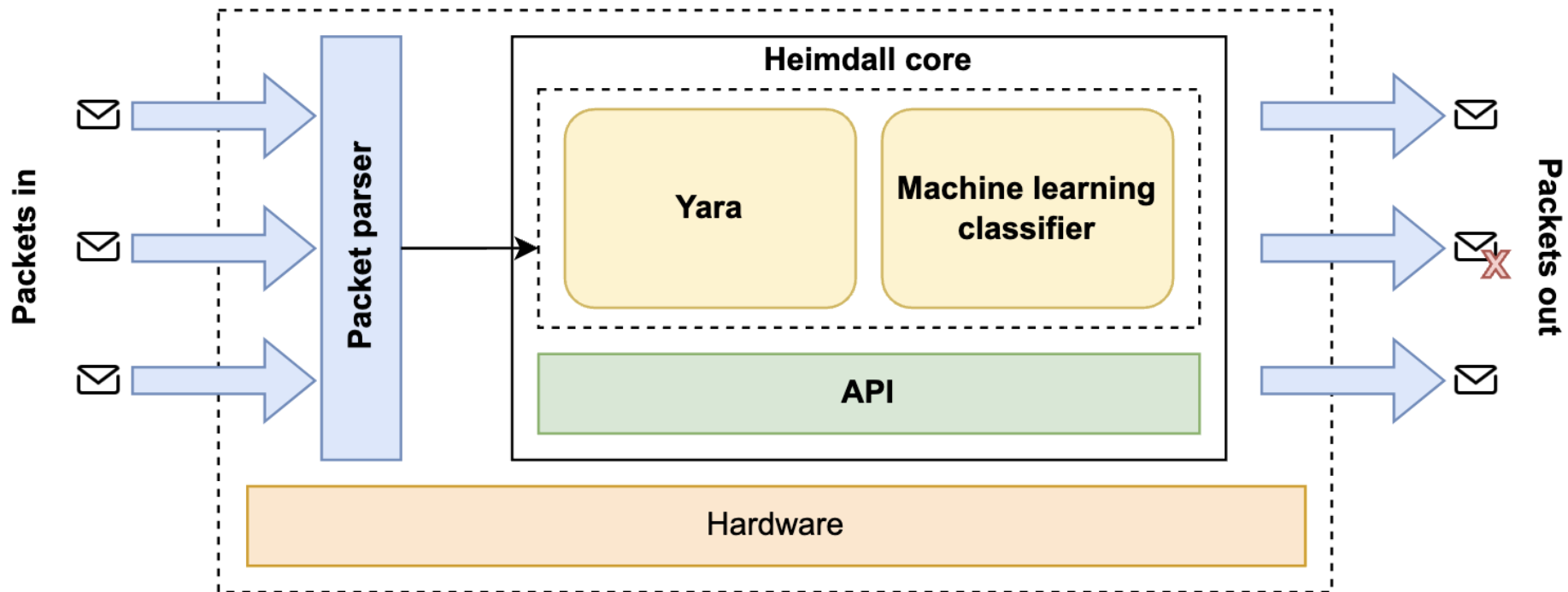
Proposta	Problema	Estratégia	Classificador	Dataset	Acurácia
(WOO; KIM; CHUNG, 2017)	Detecção de arquivos PE maliciosos	Análise de tráfego	CART	Indisponível	94.3%
(LETTERI; PENNA; GASPERIS, 2018)	Detecção de <i>botnets</i> em redes SDN	Análise de tráfego	MLP	HogZilla	96%
(CUSACK; MICHEL; KELLER, 2018)	Detecção de <i>ransomwares</i> em nível de rede	Análise de tráfego	Random Forest	Indisponível	87%
(MAEDA et al., 2019)	Detecção e isolamento de máquinas infectadas por <i>botnets</i>	Análise de tráfego	MLP	CTU-13 e ISOT	99.2%
(KHAN; AKHUNZADA, 2021)	Detecção de <i>malwares</i> em ambientes IoMT	Análise de tráfego	CNN e LSTM	Indisponível	99.83%
(MUTHANNA et al., 2022)	Detecção de intrusões em ambientes IoT	Análise de tráfego	Cu-LSTM-GRU	CICIDS2017	99.23%
(CHANG et al., 2022)	Detecção de <i>malwares</i> em ambientes IoT	Análise de tráfego	CNN	IoT-23	99%
(CHAGANTI et al., 2023)	Detecção de intrusões em ambientes IoT	Análise de tráfego	LSTM	SDNIoT e SDN-NF-TJ	97.1%

Proposta

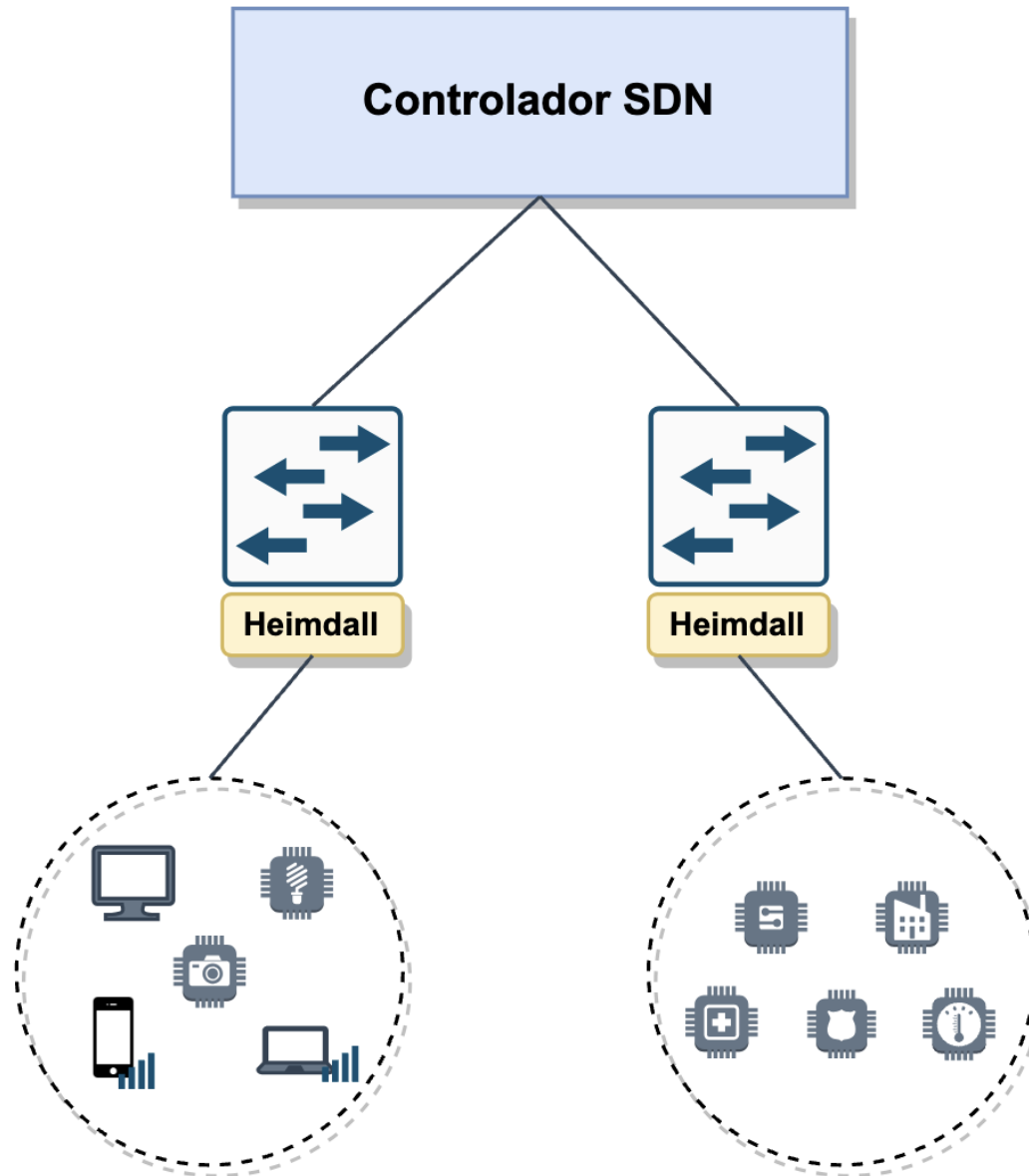
Proposta – Requisitos funcionais

- Realizar a coleta de tráfego para análise de forma agnóstica, permitindo que a solução atue independentemente dos modelos de dispositivos conectados na rede.
- Identificar padrões de malwares na rede por meio de assinaturas maliciosas.
- Realizar a classificação de fluxos maliciosos por meio de machine learning.
- Isolar dispositivos infectados do restante da rede, visando evitar a propagação de ameaças para outros dispositivos conectados.
- Prover uma API para gerenciamento e monitoramento da ferramenta.

Proposta – Arquitetura funcional

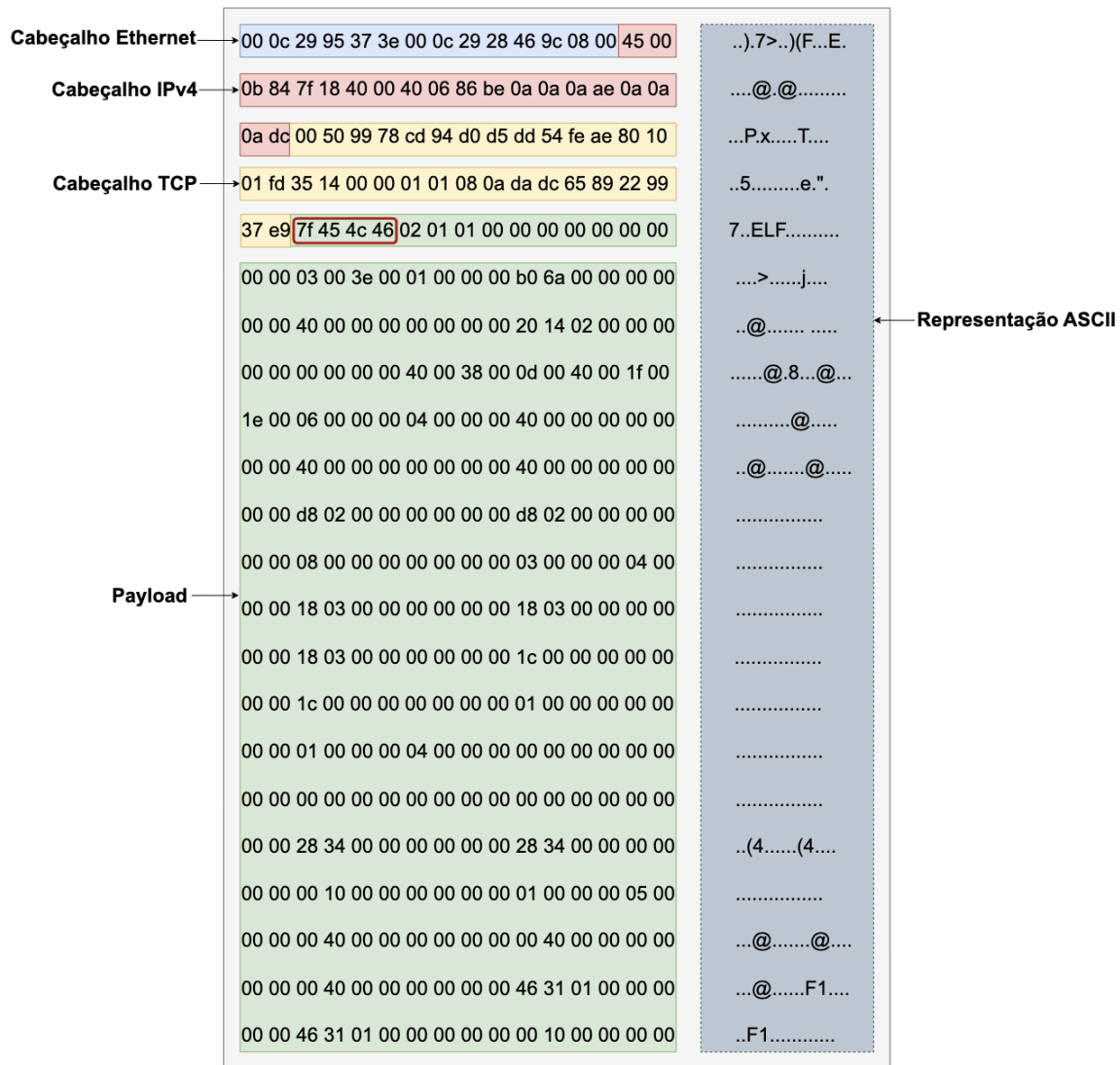


Proposta – Arquitetura funcional



Proposta – Módulos: Packet Parser

- Responsável por tratar os pacotes antes do processamento por parte do núcleo da ferramenta.
- O principal objetivo é extrair as informações relevantes para a identificação por assinaturas e classificação via machine learning.



Proposta – Módulos: Core

- Composto por três submódulos:

- **Yara:** compara o conteúdo dos pacotes com assinaturas reconhecidamente maliciosas.

```
rule NomeRegra : Tag1 Tag2 Tag3
{
  meta:
    description = "Descrição da regra"
  strings:
    $a = "Conteúdo a ser buscado no arquivo"
    $b = { 4D 5A E2 34 B1 C2 }
  condition:
    $a or $b
}
```

- **Machine learning classifier:** é responsável por analisar pacotes que não puderam ser identificados por meio das regras definidas.

Proposta – Módulos: Core

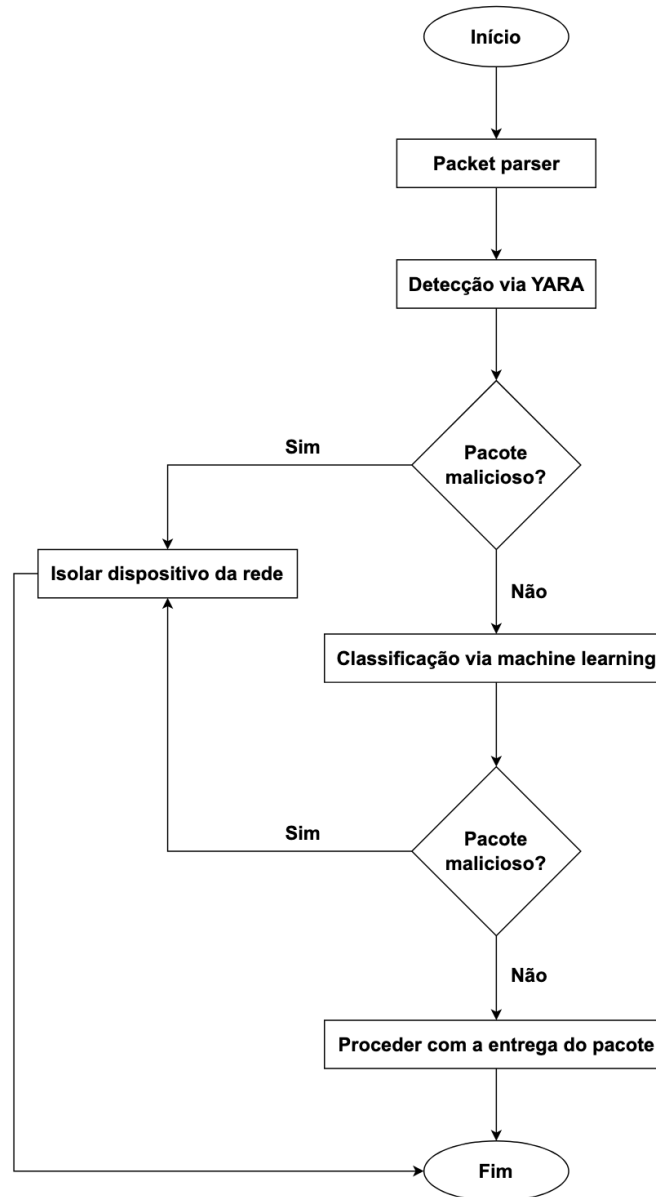
- **API:** prove recursos para o gerenciamento remoto da solução.
- Informações retornadas via JSON.

```
$ curl --request GET --url https://10.10.10.42/api/show/infected \  
--header 'x-apikey: 0c1e500ff6ea60d597ad4db206967b25 '  
[  
  {  
    "switch_uuid" : "3c26037a-3105-4ebd-a2f7-8f729d8a5461",  
    "ip_addr" : "10.10.10.50",  
    "mac_addr" : "B8:27:EB:45:AB:4F",  
    "tag" : "Mirai",  
    "time" : "2023-12-03 18:02:24"  
  },  
  {  
    "switch_uuid" : "3c26037a-3105-4ebd-a2f7-8f729d8a5461",  
    "ip_addr" : "10.10.10.51",  
    "mac_addr" : "A0:AC:69:12:AD:A7",  
    "tag" : "Mirai",  
    "time" : "2023-12-03 20:51:12"  
  }  
]
```

Proposta – Fluxo de trabalho

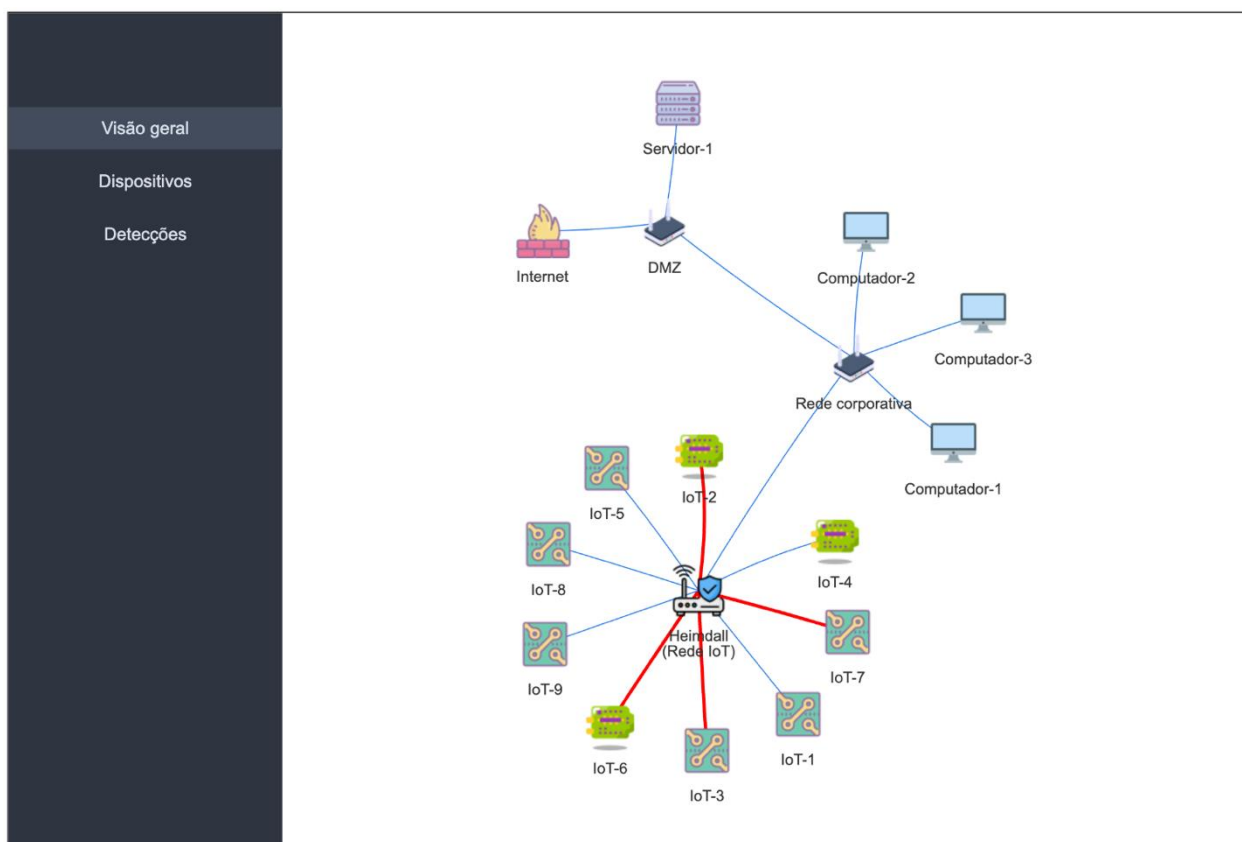
```
1: Input: packet bytes
2:
3: Begin
4:
5: payload_bytes ← Packet_Parser()
6: matches ← YARA_Analysis(payload_bytes)
7:
8: if matches then
9:     Isolate the affected device
10: else
11:     isMalicious ← Machine_Learning_Classifier(payload_bytes)
12:
13:     if isMalicious then
14:         Isolate the affected device
15:     else
16:         Forward the packet
17:
18: End
```

Proposta – Fluxo de trabalho



Proposta – Interface gráfica

- Com o intuito de facilitar a administração da solução proposta, uma interface gráfica foi desenvolvida.
- Tela principal:



Proposta – Interface gráfica

- Tela com a listagem de dispositivos:



Nome	IP	MAC	Status	Ações
IoT-1	10.10.10.51	B8:27:EB:45:AB:41	Ativo	  
IoT-2	10.10.10.52	B8:27:EB:45:AB:42	INFECTADO	  
IoT-3	10.10.10.53	B8:27:EB:45:AB:43	INFECTADO	  
IoT-4	10.10.10.54	B8:27:EB:45:AB:44	Ativo	  
IoT-5	10.10.10.55	B8:27:EB:45:AB:45	Ativo	  
IoT-6	10.10.10.56	B8:27:EB:45:AB:46	INFECTADO	  
IoT-7	10.10.10.57	B8:27:EB:45:AB:47	INFECTADO	  
IoT-8	10.10.10.58	B8:27:EB:45:AB:48	Ativo	  

Proposta – Interface gráfica

- Tela com a listagem de detecções:

Dispositivo	IP	MAC	Família	MD5	Ações
IoT-2	10.10.10.52	B8:27:EB:45:AB:42	Mirai	a6d1ac6b2f364761d673dded54914525	
IoT-3	10.10.10.53	B8:27:EB:45:AB:43	Gafgyt	1528dbfee080b4d6e45ea9ac36189b4c	
IoT-2	10.10.10.52	B8:27:EB:45:AB:42	Mirai	5dd645591b4ca486f0b8a63119bcd9f0	
IoT-6	10.10.10.56	B8:27:EB:45:AB:46	Mirai	b2356ec8d749914edcc18fd6647f3a4a	
IoT-7	10.10.10.57	B8:27:EB:45:AB:47	Hajime	d975ef8d6d5d89bbd12fbbd55871afd1	

Validação e análise

Validação e análise

- A solução é avaliada em uma plataforma de testes emulada por meio da ferramenta Mininet e com a utilização do software BMv2 P4.
- Para os cenários de avaliação propostos, é considerado que todos os dispositivos estão na mesma infraestrutura de rede.
- As avaliações foram realizadas em uma máquina com CPU Intel Core i7-11800H 2.30GHz (8 vCPUs), 32GB de RAM e sistema operacional Ubuntu Server 22.04.3 LTS 64-bits (Linux Kernel 6.2).
- O primeiro passo consiste na definição das regras YARA e treinamento dos modelos de machine learning capazes de detectar ameaças de malware a partir de um bom dataset.
- Em seguida, o modelo de maior acurácia é integrado à solução. A prova de conceito é feita considerando exemplares de malware reais.

Validação e análise

- As regras YARA utilizadas na implementação foram obtidas a partir do repositório de código aberto Yara-Rules/rules (<https://github.com/Yara-Rules/rules>).
- O dataset utilizado para treinamento dos modelos propostos neste estudo é o IoT-23. Esta base contém 20 capturas de malwares coletadas de diversos dispositivos IoT, além de 3 capturas de tráfego benigno.
- O motivo da escolha desse dataset é a sua ampla utilização em estudos anteriores. Em números totais, o dataset possui 325.307.990 registros, sendo 294.449.255 deles maliciosos.
- Foi realizado um pré-processamento nos dados para utilização na etapa de treinamento.

Validação e análise

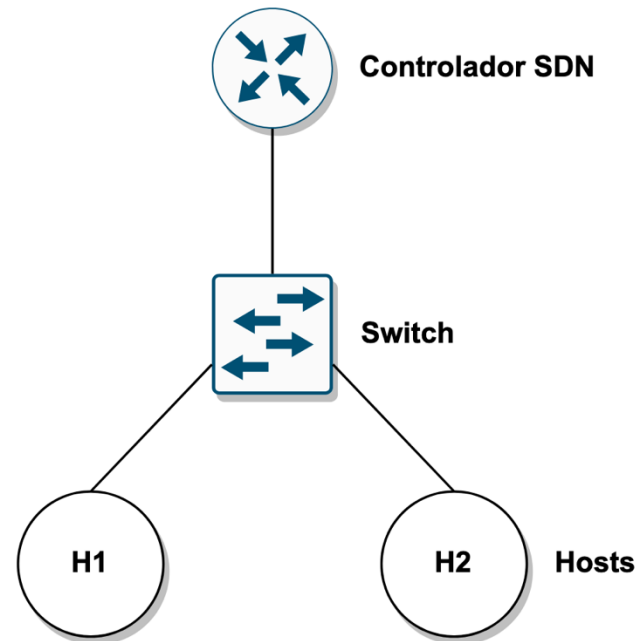
- Algoritmos implementados:

Algoritmo	Acurácia	Precisão	<i>Recall</i>	F1-Score
CNN	92.83%	0.97	0.99	0.98
Decision Tree	97.33%	0.93	0.99	0.96
Random Forest	99.33%	0.97	0.99	0.98
SVM	94%	0.91	0.93	0.92

- Com base nas métricas apresentadas, é evidente que todos os modelos atingiram um desempenho notável, com o Random Forest liderando em termos de acurácia e F1-Score.
- Diante disso, esse modelo foi escolhido para avaliação completa da ferramenta proposta.

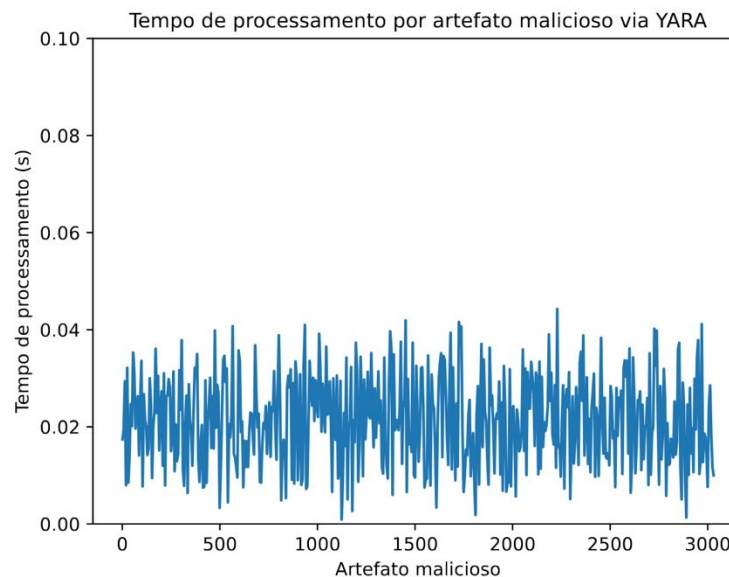
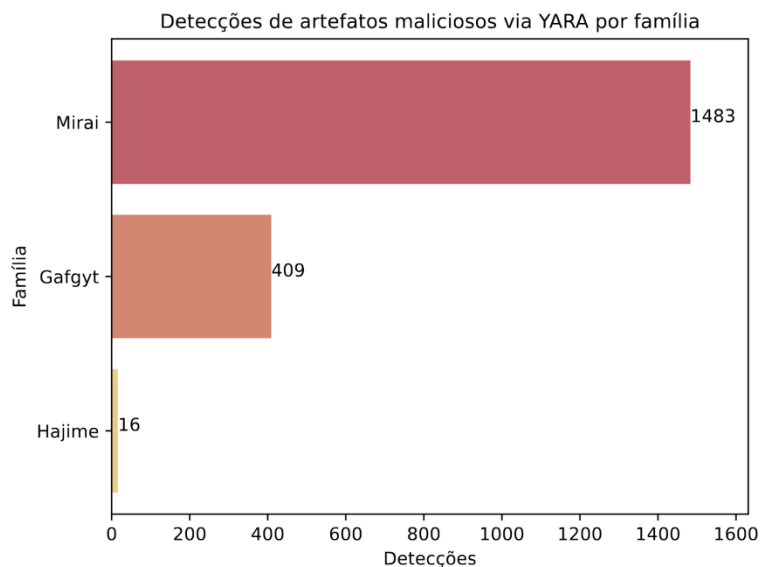
Validação e análise

- Para validar a ferramenta proposta, o modelo treinado com o algoritmo Random Forest foi acoplado à arquitetura do Heimdall em uma infraestrutura IoT habilitada por SDN.
- Com o objetivo de avaliar a capacidade de generalização do modelo e a efetividade da abordagem híbrida com o YARA, foram utilizados 3.030 exemplares de malware reais.
- Topologia utilizada:



Validação e análise

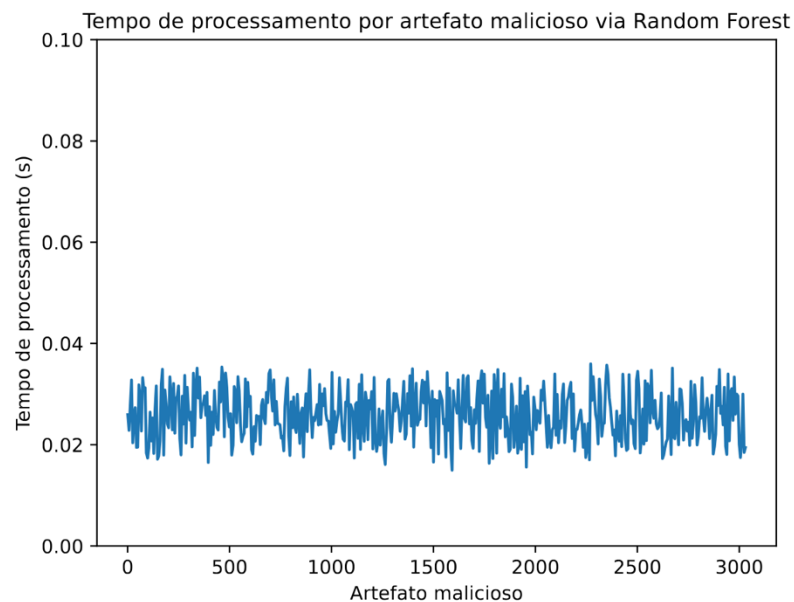
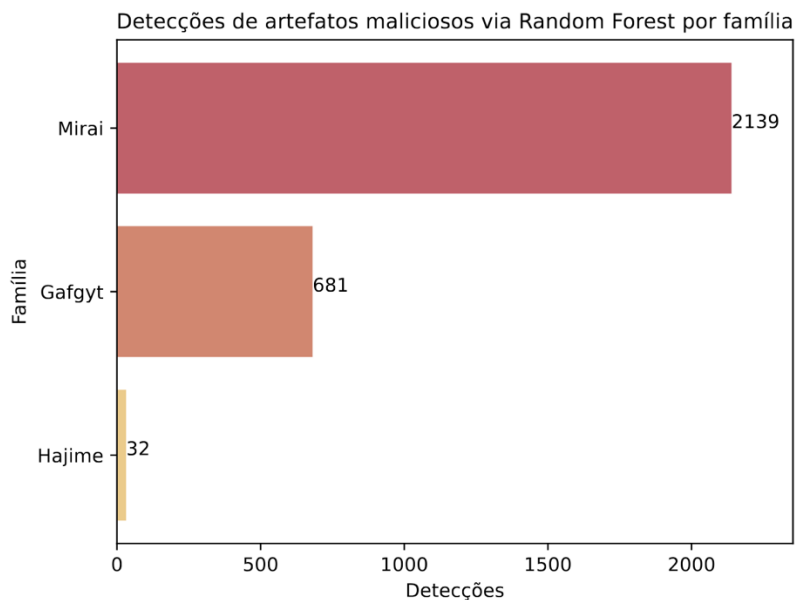
- Primeiro cenário: Apenas o módulo YARA habilitado.



Família	Quantidade	Detectados	Taxa de detecção
Mirai	2.220	1.483	66.80%
Gafgyt	754	409	54.24%
Hajime	56	16	28.57%
Total	3.030	1.908	63%

Validação e análise

- Segundo cenário: Apenas o módulo Random Forest habilitado.

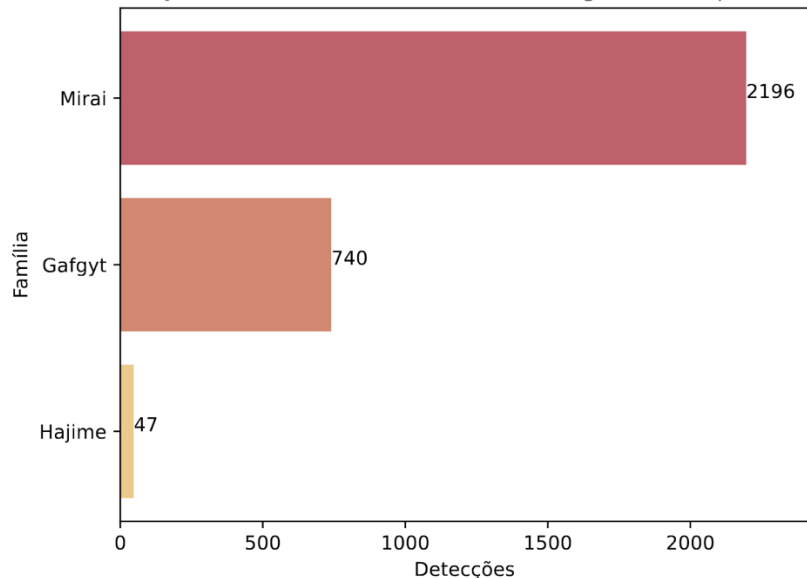


Família	Quantidade	Detectados	Taxa de detecção
Mirai	2.220	2.139	96.35%
Gafgyt	754	681	90.31%
Hajime	56	32	57.14%
Total	3.030	2.852	94.12%

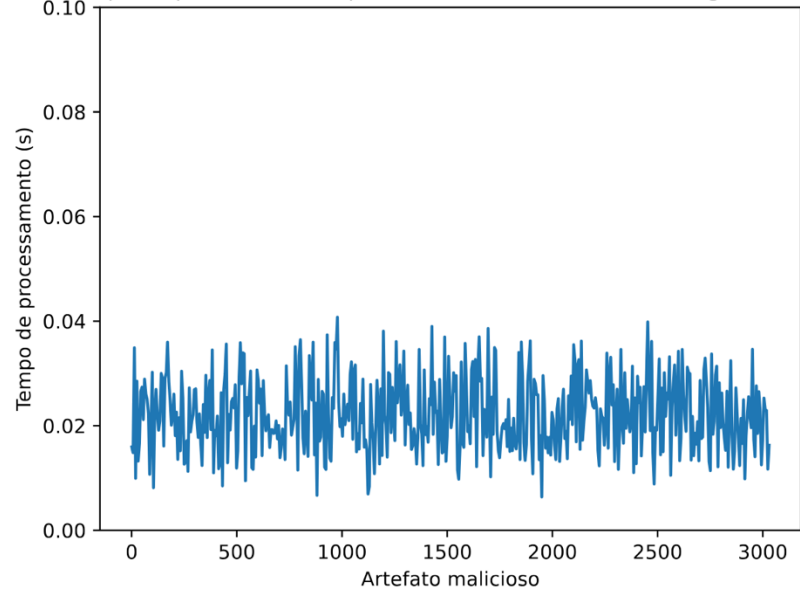
Validação e análise

- Terceiro cenário: Abordagem híbrida.

Detecções de artefatos maliciosos via abordagem híbrida por família



Tempo de processamento por artefato malicioso via abordagem híbrida



Família	Quantidade	Detectados	Taxa de detecção
Mirai	2.220	2.196	98.91%
Gafgyt	754	740	98.14%
Hajime	56	47	83.92%
Total	3.030	2.983	98.44%

Validação e análise

- Comparação entre os três cenários:

Abordagem	Tempo médio	Menor registro	Maior registro	Acurácia
YARA	0.0218s	0.0006s	0.0399s	63%
<i>Random Forest</i>	0.0259s	0.0170s	0.0349s	94.12%
Híbrida	0.0217s	0.0006s	0.0347s	98.44%

Validação e análise

- **Análise de complexidade:**

$$O(n_{yara} \cdot m_{yara} + n_estimators \cdot \log m_{rf})$$

onde n_{yara} representa o número de regras YARA utilizadas pela ferramenta;

m_{yara} a quantidade de *bytes* a serem analisados;

$n_estimators$ o número de árvores de decisão utilizadas; e

m_{rf} o número de características utilizadas.

Validação e análise - Limitações

- **Atualização das regras YARA:** É necessário realizar a atualização manualmente em cada comutador da rede.
- **Pacotes encriptados:** Lidar com pacotes criptografados (e.g., via TLS) é um desafio para ferramentas de segurança.
- **Experimentos em ambiente virtualizado:** O elevado custo de switches programáveis dificulta o acesso a tais dispositivos e, conseqüentemente, a realização de testes mais complexos.

Conclusão

Conclusão

- Este trabalho de pesquisa introduz o Heimdall, uma solução para detecção de artefatos maliciosos em ambientes IoT habilitados por SDN.
- A arquitetura da ferramenta proposta se diferencia do estado da arte por fazer o uso de uma abordagem híbrida, capaz de realizar a rápida identificação de programas maliciosos via regras YARA e a classificação deles por meio do algoritmo de machine learning Random Forest.
- Os resultados obtidos evidenciam que a abordagem híbrida proposta foi capaz de atingir uma acurácia de 98.44% e um tempo de processamento médio de 0.0217s.
- Dessa forma, o Heimdall se apresenta como uma solução viável para detecção de programas maliciosos em ambientes IoT habilitados pela tecnologia SDN.

Conclusão

- **Contribuições:**

- SOUZA, Cristian HM; ARIMA, Carlos H. A hybrid approach for malware detection in SDN-enabled IoT scenarios. **Internet Technology Letters**, p. e534.
- SOUZA, Cristian HM; ARIMA, Carlos H. Detecção de malware em ambientes IoT habilitados por SDN. In: **Anais do XV Workshop de Pesquisa Experimental da Internet do Futuro**. SBC, 2024.
- SOUZA, Cristian HM; ARIMA, Carlos H. Avaliação de algoritmos de machine learning para detecção de malware IoT no dataset IoT-23. In: **Anais do XXIV Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais**. SBC, 2024.

Conclusão

- **Produção industrial:**
 - A execução deste trabalho também resultou no registro do software intitulado "*Heimdall: Solução para detecção de artefatos maliciosos em ambientes IoT por meio de machine learning*", pelo Instituto Nacional da Propriedade Industrial (INPI), sob o número de registro **BR512024000157-3**.
 - A interface administrativa da ferramenta foi registrada no mesmo instituto com o título "*Heimdall-NG - Interface administrativa*" e número de registro **BR512024001975-8**.

Conclusão – Trabalhos futuros

- Avaliação da proposta em um ambiente real.
- Criação de uma interface gráfica para gerenciamento holístico.
- Desenvolvimento de um ambiente para análise de malware.
- Propor uma metodologia para avaliação de ferramentas focadas na detecção de malware.
- Avaliar o comportamento da ferramenta contra artefatos especializados em se evadir de defesas.
- Implementação de outros algoritmos de aprendizado de máquina.



Unidade de Pós-Graduação, Extensão e Pesquisa

