Uma abordagem de detecção de atividade maliciosa em ambientes hospitalares

Claudio A. S. Lelis, Lourenço A. Pereira, Cesar A. C. Marcondes

[1]Instituto Tecnológico de Aeronáutica
São José dos Campos/SP - Brasil

{lelis,ljr,cmarcondes}@ita.br

Abstract. *The complexity of the hospital environment and scarce resources are challenges faced by IT and cybersecurity managers. The healthcare industry is receiving increasing attention and suffering from malicious activities that lead to the breach of sensitive data. This article presents an approach to detect malicious activity in hospital environments through a classifying model based on Decision Tree, trained with data from simulations performed in a system dynamics model for the hospital environment. Experiments were conducted using the f1-score as a metric for evaluating the classification model. The results showed that the classifier tends to converge from 100 simulation replicates.*

Resumo. *A complexidade do ambiente hospitalar e os recursos escassos constituem desafios enfrentados pelos gestores de TI e de segurança cibernética. O setor da Saúde vem recebendo cada vez mais atenção, sofrendo com atividades maliciosas que levam à violação de dados sensíveis. Este artigo apresenta uma abordagem para detectar atividade maliciosa em ambiente hospitalar por meio de classificador baseado em Árvore de Decisão. O modelo foi treinado com dados de simulações executadas a partir da dinâmica de sistemas de ambiente hospitalar. Foram conduzidos experimentos de avaliação, utilizando o f1-score como métrica, com resultados satisfatórios onde o classificador tende a convergir a partir de 100 réplicas de simulação.*

1. Introdução

No dia-a-dia dos hospitais modernos, uma ampla variedade de dispositivos, sistemas e aplicativos interconectados trafegam dados, continuamente. Tais sistemas e dispositivos representam um potencial ponto de entrada para cibercriminosos que visam atacar o hospital [Fuentes and Huq 2018]. Esta situação torna a segurança cibernética um componente crítico da TI em saúde. O setor da Saúde vêm recebendo cada vez mais atenção e sofrendo com atividades maliciosas que levam à violação de dados sensíveis. Isso gera um custo para as vítimas. Estima-se que o custo médio, para cada violação de dados, tenha sido de US$6,45 milhões em 2019 [Ponemon 2019], tornando a indústria de saúde, a mais cara nesse quesito.

A área tem atraído a atenção de pesquisadores e centros de pesquisa em criar soluções para auxiliar gestores a protegerem os ativos hospitalares. Por exemplo, através de um sistema de autenticação para transmissão segura de informação médica [Bae and Han 2014], ou ainda, pelo uso de *blockchain* no armazenamento dos dados sensíveis de pacientes, médicos e os tratamentos em andamento [Pham et al. 2019]. A complexidade

do ambiente hospitalar e os recursos escassos constituem desafios enfrentados pelos gestores de TI e de segurança cibernética, na implantação das políticas de segurança exigidas por órgãos reguladores e de controle. Tal fato aumenta a vulnerabilidade nesses ambientes [Jalali and Kaiser 2018]. Apesar dos esforços, o ambiente hospitalar é complexo e capturar as nuances das diferentes políticas de segurança, refletidos nos processos que podem estar em operação, não é trivial.

Este artigo apresenta uma abordagem para detectar atividade maliciosa em ambiente hospitalar por meio de classificador baseado em Árvore de Decisão. O modelo foi treinado com dados de simulações executadas a partir da dinâmica de sistemas de ambiente hospitalar. O objetivo é obter rastreabilidade sob os fluxos e entidades desse ambiente que descrevem as atividades em análise.

O restante do artigo é composto por: seção II apresenta os conceitos de Dinâmica de Sistemas e as vulnerabilidades abordadas nos cenários de operação simulados; seção III apresenta a metodologia adotada para construir o *dataset* e, na sequência, o modelo classificador; seção IV mostra experimentos realizados, enquanto as lições aprendidas são discutidas na seção V. Finalmente, a seção VI apresenta as considerações finais.

2. Fundamentação Teórica

O entendimento de sistemas complexos e suas interações é material de estudo da Dinâmica de Sistemas. Trata-se de uma disciplina de modelagem cujos modelos utilizam o conceito de ciclos de *feedback* para organizar as informações disponíveis de um sistema. Podem ser representados através de diagramas de causa e efeito, para compreender relações causais entre entidades, ou diagramas de estoque e fluxo que são mais precisos, criando assim modelos capazes de serem simulados em um computador [Forrester 1993]. Tais modelos são capazes de descrever causas e consequências espaçadas no tempo. Isto significa que o sentido de uma mudança em uma parte de um sistema é conhecido, mas não o tamanho do efeito desta mudança em outras partes [Camiletti and Ferracioli 2002]. Assim, a aplicação de dinâmica de sistemas propicia a compreensão do comportamento genérico de um sistema (ou sistema de sistemas), baseada nos relacionamentos causais entre as variáveis que o descrevem, através de uma visão de pensamento sistêmico.

Essas características podem ser constatadas em um estudo focado na segurança cibernética de hospitais [Jalali and Kaiser 2018]. Os autores analisaram, através de entrevistas, como a dinâmica organizacional interna interage entre os hospitais para formar um sistema amplo, de segurança cibernética hospitalar, nos Estados Unidos. Os autores observaram os mecanismos principais que os hospitais usam para reduzir a probabilidade de atividade cibercriminosa. A variável que mais influencia o risco de ataque cibernético em um hospital é a complexidade do *end-point*, seguida pelo alinhamento interno das partes interessadas, sob o ponto de vista do aspecto humano.

Não são raros os relatos de ações maliciosas explorando vulnerabilidades. Por exemplo, os dispositivos médicos com uso intensivo de Internet das Coisas (IoT) possuem sistema operacional embarcado com baixo poder computacional e criptográfico. Assim, são facilmente alvos de *malware* injetáveis tornando-os dispositivos de uma *botnet* hospitalar, que poderia atacar e interromper o fluxo de informação de monitoração de vários pacientes em estado grave, usando técnica de negação de serviço distribuída (DDoS, em inglês *Distributed Denial of Service*). Sem a devida monitoração em tempo real, existe

uma ameaça iminente à vida dos pacientes, devido a saturação de tráfego dos *bots* para o sistema de registros médicos (EHR). Outro caso de ação maliciosa pode acontecer pela clonagem de endereços físicos de *Media Access Control* (MAC) em nível da rede sem fio institucional, fazendo com que dispositivos médicos e EHR troquem mensagens, através do agente malicioso, que pode enviar um comando adulterando o regime de infusão intravenosa de um medicamento. Este ataque é conhecido como MinM (*Man-in-the-Middle*).

2.1. Modelo de Dinâmica de Sistemas para Ambiente Hospitalar

O presente artigo está inserido no contexto de um estudo anterior [Lelis et al. 2020], no qual aspectos operacionais de um hospital foram modelados usando Dinâmica de Sistemas. O modelo foi gerado com base na especificação estrutural técnica de um hospital real, bem como, as políticas organizacionais implantadas no mesmo.

Foi realizado um mapeamento no qual cada quarto do hospital foi modelado como um conjunto de 5 dispositivos médicos que monitoram o paciente e constituem uma rede próxima ao corpo do paciente (Body Area Networks - BAN) que trafega os dados através de um ponto de acesso wifi, até os dados chegarem ao sistema EHR. Também foram considerados leitos de unidade de terapia intensiva UTI no qual o dobro (10) de dispositivos monitoram os pacientes e contam com o dobro de prioridade no atendimento das requisições vista a criticidade de sua operação. Cada entidade mapeada foi modelada utilizando a notação de estoque representando a fila de requisições (*buffer*). Tais entidades são: (i) EHR; (ii) wifi; (iii) dispositivos médicos e BAN dos quartos de internação; (iv) dispositivos médicos e BAN dos leitos de UTI; (v) computadores dos quiosques de atendimento de médicos e enfermeiros. De mesmo modo, a interação entre as entidades foram modeladas utilizando a notação de fluxo que possui origem e destino indicando as requisições que saem da origem e chegam ao destino.

Os autores mostraram uma prova de conceito exercitando o modelo gerado, em 3 cenários de operação: operação normal, sob ataque MinM e sob ataque DDoS. O resultado das simulações a partir desta modelagem descrita, podem ser armazenadas como ocorrências em função do tempo de simulação. Esta característica facilitou o processo de geração dos dados, detalhado na próxima seção.

2.2. Trabalhos relacionados

A segurança cibernética na área de saúde vem recebendo a atenção dos pesquisadores, mostrando as diferentes superfícies de ataques em hospitais e clínicas e por vezes, focados em aspectos específicos da infraestrutura hospitalar.

Um exemplo disso está associado ao fluxo das informações médicas que precisam ser protegidas de possíveis atacantes intermediários. Diante disso, [Bae and Han 2014] apresentaram uma abordagem de sistema de autenticação para a transmissão segura de informação médica.

No escopo da segurança do sistema EHR, os autores em [Pham et al. 2019] aplicaram uma abordagem baseada em *blockchain* para o armazenamento seguro dos dados sensíveis de pacientes, médicos e os tratamentos em andamento.

3. Metodologia

Para alcançar o objetivo de detectar atividades maliciosas em ambientes hospitalares, utilizamos o modelo de dinâmica de sistemas mencionado na seção anterior.

Para representar o comportamento do fluxo de cada BAN, a metodologia adotada consistiu de uma série de passos que se iniciaram na geração pseudo-aleatória de distribuições estatísticas log-normal.

3.1. Geração do Dataset

Inicialmente foi estabelecido um conjunto de sementes para a geração pseudo-aleatória das distribuições log-normal. Cada distribuição representa um dispositivo médico que compõe uma BAN. Com isso foi possível preparar o ambiente para executar o modelo repetidas vezes e com distribuições diferentes por rodada.

Para garantir que cada ocorrência de passagem de tempo, é completa e independente, foi criada uma coluna extra para cada estoque no modelo para representar o valor do estoque em $t - 1$.

As simulações executadas do modelo de dinâmica de sistemas resultaram em um conjunto de dados que representa 200 réplicas para cada um dos 3 cenários de operação (operação Normal, Ataque DDoS e Ataque MinM). Considerando que cada execução refletiu 1800 passos de simulação (ocorrências), envolvendo 47 atributos. Assim, o conjunto total de dados disponíveis para a fase de geração do Dataset, e posterior aplicação de uma técnica de classificação, é de $200 \cdot 3 \cdot 1800 \cdot 47 = 50.76$ milhões de dados, contendo apenas valores numéricos e contínuos.

3.1.1. Análise de Convergência das réplicas de simulação

Inicialmente, foram selecionados os atributos individualmente e para cada cenário de operação foi calculada a variância dos valores obtidos em cada réplica. Como resultado, foi obtido um conjunto de dados de variâncias ($200 \cdot 141 = 28200$).

Foi observado nos gráficos gerados com esses valores uma alta variação. Uma justificativa para isto pode estar relacionada a natureza dos elementos do modelo de Dinâmica de Sistemas.

Assim, foram calculadas as variâncias acumuladas. Isto propiciou a identificação visual, nos gráficos gerados, de um ponto a partir do qual os valores pareciam convergir. Um exemplo é mostrado na Figura 1 que representa a variância acumulada entre as réplicas. É possível observar que a partir da réplica 100 o comportamento tende a estacionar. No entanto, o mesmo não pode ser aplicado ao comportamento da variância ao analisar um fluxo entre uma BAN e o Wi-Fi. A Figura 2 retrata essa situação em que o comportamento da variância acumulada não parece convergir.

Devido a essa heterogeneidade de comportamentos e a convergência não se verificar em todos os elementos, optou-se por não reduzir o número de réplicas a serem usadas no Dataset Inicial resultante.

3.2. Análise de dados

A etapa anterior foi responsável por gerar o Dataset Inicial, a partir de um input controlável (as distribuições lognormal). No entanto, os diferentes fatores que influenciam os valores resultantes de cada entidade, ao longo dos ciclos de simulação, caracterizam a

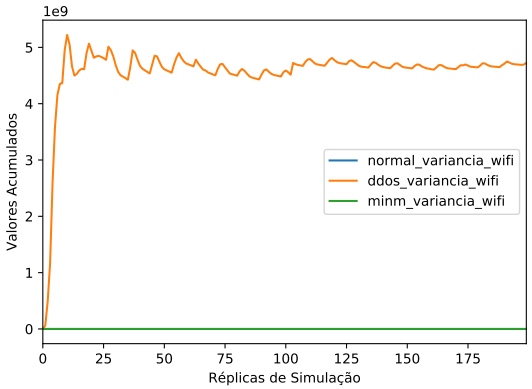

Figura 1. Variâncias acumuladas para atributo wifi

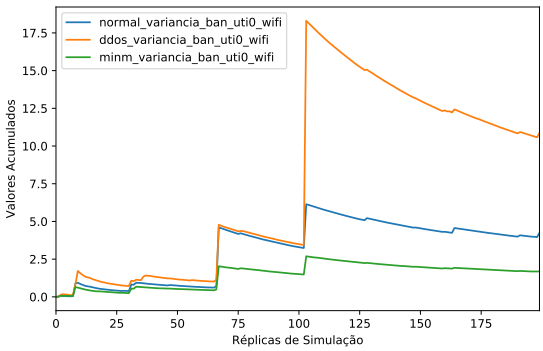

Figura 2. Variâncias acumuladas para atributo que indica fluxo entre BAN e Wifi

visão sistêmica. Por isso, é incerto que o comportamento inicial parametrizado para cada entidade foi mantido até o término do processo de simulação do modelo dinâmico.

Diante disso, faz-se necessária uma análise prévia para tentar compreender como os dados descrevem as consequências das sucessivas simulações realizadas. Assim, o objetivo nesta etapa é explorar e preparar os dados tentando reduzir o Dataset, para balancear o custo de processamento com o resultado a ser obtido na etapa de classificação.

A Figura 3 esquematiza a metodologia adotada, detalhando a sequência de passos seguidos para a análise descritiva dos dados. Inicialmente, a remoção de ocorrências duplicadas. Em seguida, a seleção de características que será detalhada na próxima seção, bem como a geração dos modelos iniciais sem parametrização. No Passo 4, a poda posterior da árvore gerada, utilizando o algoritmo *Cost-Complexity Pruning* [Steinberg and Colla 2009], possibilitando por fim, a geração dos modelos com parametrização.

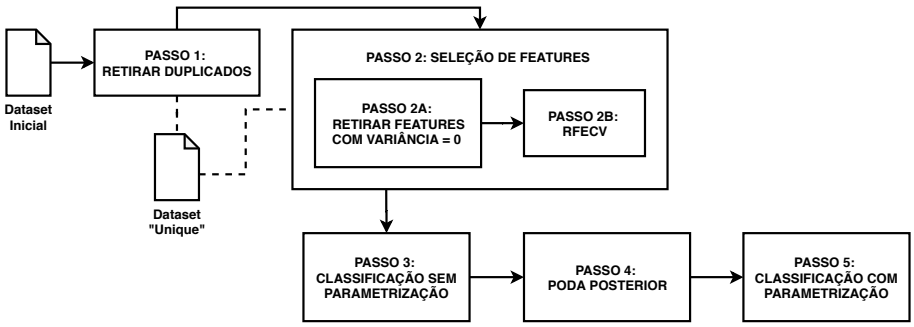

Figura 3. Metodologia

O Dataset Inicial é composto de 47 atributos. Destes, 46 representam os fluxos e estoques do modelo de dinâmica de sistemas simulado. O último atributo é o alvo que representa o cenário de operação.

3.2.1. Seleção de Características

Inicialmente foi aplicada uma abordagem simples para a seleção de características (também referenciado como *features* o qual será utilizado deste ponto em diante) baseada no valor da variância, removendo todas as *features* cuja variação não atinge algum limite. Por padrão, o algoritmo remove todas as *features* de variação zero, ou seja, que possuem o mesmo valor em todas as amostras. Dessa forma, não foram constatadas *features* com variação zero. Assim, nenhuma foi removida e aplicou-se uma abordagem mais detalhada, chamada RFECV (sigla do inglês que significa Eliminação Recursiva de *Features* com Validação Cruzada) [Guyon et al. 2002].

A ideia da eliminação recursiva de features é selecionar considerando recursivamente, conjuntos cada vez menores de features. Inicialmente, é utilizado um estimador que atribui pesos às features, e este é treinado no conjunto inicial de features. A importância de cada feature é obtida através de uma função oferecida pelo estimador escolhido (que retorna o coeficiente, ou o grau de importância). Em seguida, as features menos importantes são removidas do conjunto atual de features. Esse procedimento é repetido recursivamente até que o número desejado de features a serem selecionados seja atingido. O RFECV executa o algoritmo de Eliminação Recursiva de Features em um loop de validação cruzada para encontrar o número ideal de features a serem selecionadas.

Como resultado, foi identificado o número ótimo de 5 features a serem selecionadas. São elas: ehr, $ban\_intern0$, ban_uti1, ban_uti0_wifi, $ehr\_wifi\_\_uti1$. Respectivamente, ehr é o estoque que representa o sistema EHR que armazena os dados clínicos dos pacientes e procedimentos médicos, $ban\_intern0$ e ban_uti1 são os estoques que representam uma BAN em operação em um quarto de internação e outra BAN em operação em um leito de UTI. As features ban_uti0_wifi e $ehr\_wifi\_\_uti1$ fazem referência aos fluxos: (1) entre a BAN de UTI e o ponto de acesso wifi; e (2) entre o EHR e outra BAN de UTI, passando pelo ponto de acesso wifi.

3.3. Classificação

Diz-se que um programa aprende a partir da experiência *E*, em relação a uma classe de tarefas T, com medida de desempenho P, se seu desempenho em *T*, medido por *P*, melhora com E. Dessa forma, definimos:

- Problema: Detectar atividade maliciosa em ambientes hospitalares.
- Tarefa T: classificação multi classes dos cenários simulados. As classes são: Operação Normal (0), Ataque DDoS (1) e Ataque MinM (2)
- Medida de desempenho P: F1-score da medida das classificações.
- Experiência E: Réplicas de simulação do modelo dinâmico hospitalar.

Considerando o objetivo inicial do trabalho, de obter rastreabilidade entre os fluxos e entidades do ambiente, foi escolhida a técnica de classificação baseada em árvore de decisão. Em todo o processo foi utilizada a biblioteca *scikit-learn* [Pedregosa et al. 2011], para a linguagem Python.

Árvore de Decisão é um algoritmo de machine learning tipo caixa branca. Ele compartilha a lógica interna de tomada de decisão, que não está disponível no tipo de algoritmo de caixa preta, como a Rede Neural. A complexidade do tempo das árvores de decisão é uma função do número de registros e do número de atributos nos dados

fornecidos. A árvore de decisão é um método livre de distribuição ou não paramétrico, que não depende de premissas de distribuição de probabilidade. O estimador utilizado no algoritmo RFECV também foi a arvore de decisão. As classificações, a poda posterior e as parametrizações serão detalhadas na próxima seção.

4. Experimentos

Esta seção apresenta o experimento conduzido. Segundo a abordagem GQM (*Goal, Question, Metric*) [Basili and Weiss 1984], o objetivo pode ser enunciado como: **Analisar** o classificador gerado **a fim de** verificar a viabilidade de uso **com relação à** classificar a ocorrência de atividades maliciosas **do ponto de vista** dos cenários de operação (normal, MinM e DDoS) **no contexto de** cibersegurança na infraestrutura hospitalar.

Os experimentos utilizaram o Dataset resultante da exclusão de ocorrências duplicadas e da seleção de features, ou seja, 5 atributos, além do atributo alvo, contendo dados numéricos e contínuos. Como já mencionado, o número de réplicas de simulação não foi reduzido previamente, assim, a estratégia adotada foi de conjuntos parciais de réplicas (a saber: 30 réplicas, 100 e 200 réplicas).

Com a finalidade de avaliar as predições feitas pelo classificador foi aplicado o método de validação cruzada 10-fold e utilizada a métrica *f1-score*, que consiste em uma medida da acurácia no teste. O *f1-score* pode ser interpretado como uma média ponderada do *precision* p [1] e o *recall* r [2] , onde o *f1-score* atinge seu melhor valor em 1 e pior valor em 0. A contribuição relativa de p e r para o *f1-score* é igual. A fórmula é dada por:

$$F1 = 2 \cdot (p \cdot r)/(p + r) \tag{1}$$

4.1. Classificação Sem Parametrização

A Tabela 1 mostra o valor médio obtido do f1-score nas 10 execuções da validação cruzada 10-fold para geração do classificador de árvore de decisão. Foram testados dois critérios para o corte entre atributos: (1) o índice Gini e (2) a entropia que utiliza a abordagem de ganho de informação.

Tabela 1. Valores médios de f1-score - sem parametrização

	Réplicas	F1-Score	MÁX Altura	Nro de Nós Folha
Índice Gini	30	0.962559	30.0	475.0
Entropia	30	0.962397	28.0	480.2
Índice Gini	100	0.949587	40.0	1713.4
Entropia	100	0.949028	45.0	1842.2
Índice Gini	200	0.949425	50.0	3469.6
Entropia	200	0.948993	51.0	3613.3

Apesar dos bons resultados atingidos, o classificador resultante sofreu sobreajuste (overfitting) gerando, assim, uma arvore complexa com altura máxima de 51 nós e média de 3613.3 nós folha, para o caso de 200 réplicas.

[1] número de resultados positivos corretos dividido pelo número de todos os resultados positivos retornados pelo classificador

[2] número de resultados positivos corretos dividido por o número de todas as amostras que deveriam ter sido identificadas como positivas

Diante disso, para reduzir a complexidade da árvore gerada, optou-se por realizar uma poda posterior com o objetivo de minimizar a altura, maximizando a acurácia. Foi aplicada a técnica *Cost-Complexity Pruning* (CCP) [Steinberg and Colla 2009] que fornece uma opção para controlar o tamanho de uma árvore. Essa técnica de remoção pode ser parametrizada por um parâmetro $alpha$.

Na prática, o CCP encontra recursivamente o nó da árvore com a ligação mais fraca. Este link mais fraco é caracterizado por um $alpha$ no qual os nós com o menor $alpha$ são podados primeiro. À medida que o $alpha$ aumenta, mais ramificações são podadas. Consequentemente, o número de nós e a profundidade da árvore diminui e aumenta a impureza total de suas folhas.

O classificador gerado com 200 réplicas e o critério índice Gini foi utilizado na aplicação do CCP e análise dos valores de $alpha$. Para medir a acurácia e permitir relacionar com a altura da árvore, foi realizada a divisão do Dataset em conjunto de treinamento e conjunto de teste, o primeiro para treinar um classificador com os diferentes valores de $alpha$ e o segundo para testar a classificação.

Tabela 2. Valores de Alpha com os parâmetros resultantes

CCP Alpha	Máx. Altura	Nro Nós Folha	Acurácia Conj. Teste
0.000000	51	6407	0.958329
0.000001	48	3937	0.958550
0.000003	46	2879	0.958214
0.000004	43	2441	0.958028
0.000005	42	1911	0.957718
0.000039	14	79	0.956922
0.000042	10	59	0.956736
0.000050	9	53	0.956692
0.000088	8	41	0.956489
0.000395	7	23	0.955630
0.001352	6	19	0.954808
0.001664	5	13	0.951623
0.004412	3	9	0.947340
0.005736	2	7	0.944713
0.190998	1	3	0.736605
0.356751	0	1	0.442704

A Tabela 2 reúne os valores de altura máxima, número de nós folha e acurácia do conjunto de teste, a partir de alguns dos valores do parâmetro $alpha$ com o resultado consolidado. Para facilitar a visualização, a Tabela 2 mostra apenas os maiores e menores valores encontrados de $alpha$ e para valores que geraram alturas repetidas, foi escolhido o menor número de nós folha, e consequentemente o pior caso de acurácia.

4.2. Classificação Com Parametrização

A Tabela 3 mostra o valor médio obtido do f1-score nas 10 execuções da validação cruzada 10-fold para geração do classificador de árvore de decisão. Os valores resultantes *com parametrização* foram gerados também considerando os critérios de corte índice gini e entropia. Baseado na análise do CCP, foi escolhido o valor de $alpha = 0.001352$ que

consequentemente define o critério de parada da profundidade para 6, ou seja, no máximo a arvore terá altura de 6.

Tabela 3. Valores médios de f1-score - com parametrização

	Réplicas	F1-Score	MÁX Altura	Nro de Nós Folha
Índice Gini	30	0.955420	5.0	7.0
Entropia	30	0.957738	5.0	9.0
Índice Gini	100	0.944968	5.0	8.0
Entropia	100	0.945744	6.0	11.0
Índice Gini	200	0.944617	6.0	10.0
Entropia	200	0.944454	6.0	11.0

A Figura 4 mostra a relação entre a acurácia do conjunto de teste e a altura da árvore que é definida através do valor de $alpha$. Em destaque está a altura no valor 6 que apresenta um melhor balanço entre os fatores: máximo de altura, quantidade de nós folha e acurácia.

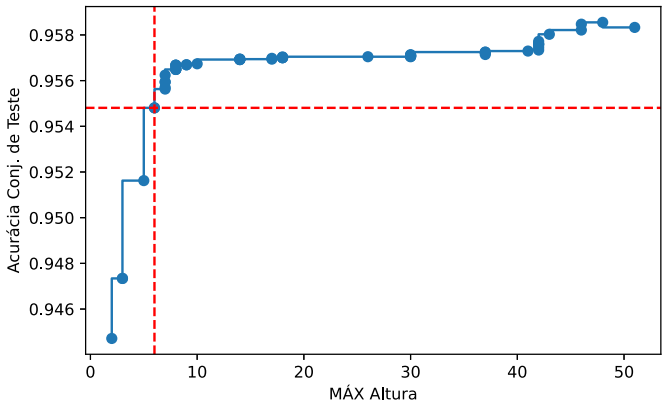

Figura 4. Altura máxima vs. Acurácia Conjunto de Teste

4.3. Análise dos Resultados

A diferença entre os resultados com 30, 100 e 200 réplicas pode ser explicada pelo fato de que 200 réplicas capturaram mais detalhes das simulações realizadas. Esta visão se reflete em uma árvore com mais nós folha e maior altura a medida que se aumentou o número de réplicas. Diante disso, os valores com 200 réplicas são discutidos em detalhes.

Ao compararmos o melhor caso de 200 réplicas e sem parametrização e o melhor caso de 200 réplicas com parametrização, o f1-score sofreu queda de 0.4782% no caso do uso da entropia e queda de 0.5064% no caso do índice gini, com a parametrização. Esta queda é esperada visto que a aplicação do CCP aumenta a impureza nos nós folha e reduz a precisão.

No entanto, houve a redução na complexidade da árvore. Sem a parametrização, são necessárias no máximo 51 (caso da entropia) ou 50 (caso do índice gini) comparações para a árvore alcançar um nó folha (nó de decisão). Com a parametrização são necessárias no máximo, apenas 6 comparações. Trata-se de uma redução de 88.23% na altura. Por

fim, o número de nós folha no caso do índice gini foi reduzido de uma média de 3469.6 nós, para apenas 10. Esta redução é ainda maior ao comparar os valores no caso da entropia, que reduziram de 3613.3 para apenas 11 nós.

Devido à porcentagem de queda do f1-score ser menor e a redução na altura e número de nós folha ser maior no caso da entropia, esta foi a escolhida como critério de corte na geração da arvore de decisão resultante, mostrada na Figura 5.

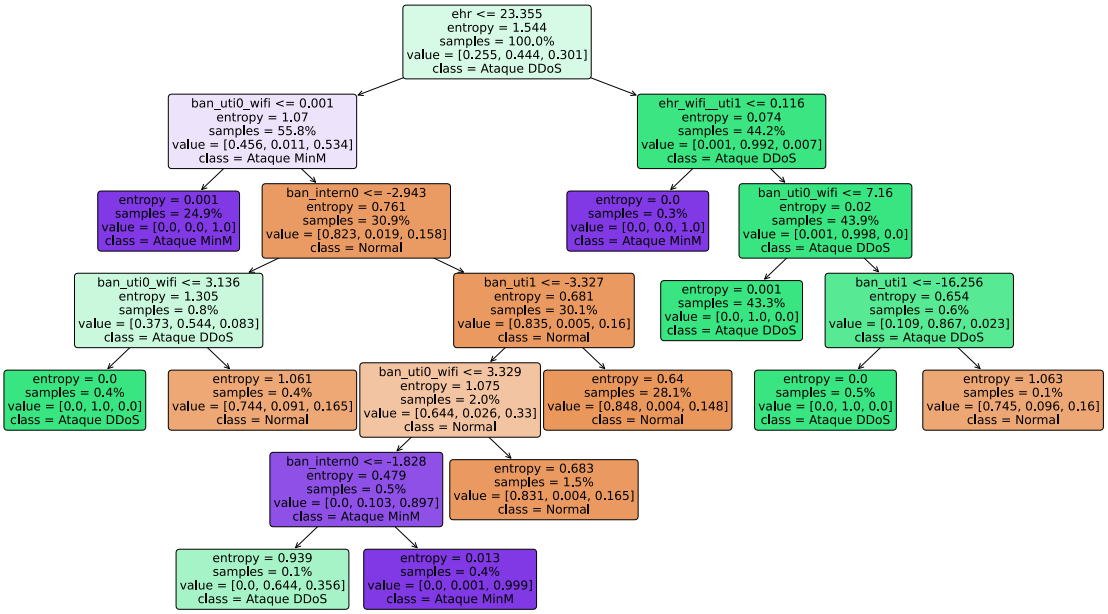

Figura 5. Árvore de Decisão resultante

4.3.1. Interpretando a árvore gerada

A árvore pode ser analisada para tomada de decisão em mudanças nos níveis operacional e organizacional de ambientes hospitalares. Neste caso, é prudente interpretar a árvore considerando famílias de dispositivos. Por exemplo, o estoque de nome ban_uti1 representaria a família de dispositivos associada às BANs em funcionamento nos leitos de UTI, evitando assim, a análise ingênua na qual o monitoramento seria direcionado especificamente para o dispositivo ban_uti1. Envolver a família de dispositivos em uma política de segurança e proteção, permite que o regime de operação mude ao ocorrer um ataque e toda a família dos dispositivos de UTI seja protegida.

Ao observar a árvore gerada, destacada na Figura 5 notamos as regras que conduzem à classe *"Ataque MinM"* (nós folha na cor roxa). Como exemplo, sejam as Regras 1, 2 e 3 que conduzem à detecção de um ataque MinM e a Regra 4 que conduz à detecção de um ataque DDoS, mostradas a partir das regras, classificando os ataques, a seguir:

- R1: Se $(ehr > 23.36)$ e $(ehr\_wifi\_\_uti1 <= 0.12)$ Então classe = MinM
- R2: Se $(ehr <= 23.36)$ e $(ban\_uti0\_wifi <= 0.00)$ Então classe = MinM
- R3: Se $(ehr <= 23.36)$ e $(ban\_intern0 > -2.94)$ e $(ban\_uti1 <= -3.33)$ e $(ban\_uti0\_wifi <= 3.33)$ e $(ban\_intern0 > -1.83)$ Então classe = MinM

- R4: Se $(ehr <= 23.36)$ e $(ban\_intern0 > -2.94)$ e $(ban\_uti1 <= -3.33)$ e $(ban\_uti0\_wifi <= 3.33)$ e $(ban\_intern0 <= -1.83)$ Então classe = DDoS

Considerando as Regras 1 e 2 é preciso que haja um monitoramento constante no EHR. Além disso, é possível definir níveis de alerta para os fluxos entre a família de BANs de UTI e o EHR (através do wifi) para detectar os valores dos fluxos e consequentemente um ataque MinM em curso.

De modo semelhante, considerando as Regras 3 e 4 o monitoramento do EHR, da família de BANs de UTI e do fluxo com o wifi é mantido, acrescentando a família de BANs dos quartos de internação. Neste ponto a vigilância deve ser ainda maior já que as BANs de internação vão servir como balizadores também para detectar um ataque DDoS.

Esse procedimento de análise pode ser repetido para todas as Regras expostas pela árvore resultante, auxiliando na adequação do modelo dinâmico, bem como, nas políticas de segurança implementadas.

5. Lições Aprendidas

O modelo de Dinâmica de Sistemas utilizado é constituído de fluxos positivos que indicam o sentido da informação e fluxos negativos que indicam o retorno da informação. Embora haja a possibilidade de controlar os valores de entrada do modelo como realizado nos experimentos, no entanto, os desdobramentos e os valores resultantes estão associados a variáveis aleatórias, sendo assim de baixa previsibilidade. Diante disso, são monitorados os comportamentos de fluxos e estoques com valores positivos, bem como, negativos.

O modelo de classificação é treinado com o resultado da simulação que indica os valores numéricos de cada entidade e fluxo do ambiente. Isso facilita a análise da árvore de decisão gerada e principalmente do retorno das regras para refinar o modelo, e assim, refinar as políticas operacionais implantadas. A convergência do aprendizado pode ser um indício da complexidade do meio e do ambiente hospitalar. Sendo heterogêneo e de complexa previsão.

A árvore de decisão gerada pode apresentar um usuário como atributo de interesse diante do padrão identificado. Isso ocorreu no caso do atacante e seus fluxos. O atacante é mais um usuário e a tomada de decisão de monitorar representa um desafio quando alinhado a aspectos da políticas dos usuários e direitos de privacidade. Uma abordagem pode ser a retirada do processo de aprendizado, bem como, o fluxo gerado em direção ao usuário. Desta forma, estrutura-se a segurança interna, filtrando ações maliciosas.

O conhecimento dos ataques, e respectivos processos, permitem planejar o monitoramento e as contramedidas de defesa. Assim a aplicação de um modelo classificador é uma ferramenta interessante propiciando *insights* para o refino do próprio modelo dinâmico. Enfim, o processo é iterativo e incremental de forma que o conhecimento é adquirido gradativamente com as simulações.

6. Considerações Finais

Este estudo apresentou uma abordagem para detectar atividade maliciosa em ambientes hospitalares, utilizando como técnica de classificação árvores de decisão. Os resultados do experimento conduzido mostraram que a métrica usada na avaliação do modelo de

classificação f1-score, tende a convergir a partir da réplica 100. Foi mostrado também a árvore de decisão gerada a partir de uma parametrização do modelo.

O objetivo é que o conhecimento adquirido na detecção de atividades maliciosas seja aplicado na melhoria das políticas operacionais de segurança e processos operacionais de TI em ambientes hospitalares.

Como trabalhos futuros pretende-se aplicar outras técnicas de classificação mas aliando com as regras de classificação. Além disso, aumentar a capacidade do modelo de dinâmica de Sistemas em representar outros cenários de operação.

Referências

Bae, W.-S. and Han, K.-H. (2014). An authentication system for safe transmission of medical information in u-health environment. *International Journal of Applied Engineering Research*, 9(20):7909–7918.

Basili, V. R. and Weiss, D. M. (1984). A methodology for collecting valid software engineering data. *IEEE Transactions on software engineering*, (6):728–738.

Camiletti, G. G. and Ferracioli, L. (2002). The use of semiquantitative computational modelling in the study of predator-pray system. *X International Organization for Science and Technology Education, 2002, Foz do Iguaçu*.

Forrester, J. W. (1993). System dynamics and the lessons of 35 years. In *A systems-based approach to policymaking*, pages 199–240. Springer.

Fuentes, M. R. and Huq, N. (2018). Securing connected hospitals. https://documents.trendmicro.com/assets/rpt/rpt-securing-connected-hospitals.pdf, accessed: 10-June-2019.

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3):389–422.

Jalali, M. S. and Kaiser, J. P. (2018). Cybersecurity in hospitals: a systematic, organizational perspective. *Journal of medical Internet research*, 20(5):e10059.

Lelis, C. A. S., de Oliveira Filho, S. R. A., and Marcondes, C. A. C. (2020). Towards hospital dynamics model in the age of cybercrime. In *17th International Conference on Information Technology–New Generations (ITNG 2020)*, pages 469–475. Springer.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pham, H., Tran, T., and Nakashima, Y. (2019). A secure remote healthcare system for hospital using blockchain smart contract. *2018 IEEE Globecom Workshops, GC Wkshps 2018 - Proceedings*.

Ponemon, L. (2019). What's new in the 2019 cost of a data breach report. https://securityintelligence.com/posts/whats-new-in-the-2019-cost-of-a-data-breach-report/ , accessed 10 June 2019.

Steinberg, D. and Colla, P. (2009). Cart: classification and regression trees. *The top ten algorithms in data mining*, 9:179.