Automatic Challenge Generation for Teaching Computer Security

Ricardo de la Rocha Ladeira
Departamento de Ensino
Instituto Federal Catarinense
Campus Blumenau
ricardo.ladeira@ifc.edu.br

Rafael R. Obelheiro
Departamento de Ciência da Computação
Universidade do Estado de Santa Catarina
Campus Joinville
rafael.obelheiro@udesc.br

***Abstract*—Computer Security is an increasingly important area, considering the sophistication and increase of threats present in the digital world. The need for information protection contrasts with the lack of professionals and the limited space dedicated to the area in Information Technology (IT) courses. Games and competitions have been used to motivate students of Computing to improve their practical knowledge on the subject and also to foster the interest of potential students 'and professionals in Security. The creation of these games requires specialized knowledge to develop new problems, since the novelty of these games is important to reach the desired level of difficulty and to ensure competitiveness. This work proposes the use of randomization to generate problems and entire competitions in an automated way, obtaining exclusive instances of problems for each player. As proof of concept, a tool for generating challenges was developed to evaluate the proposal. Competitions with automatically generated problems were promoted, which included students of undergraduate courses and professional qualification in Computing, in two different institutions. The performance in the competitions and the perception of satisfaction, interest and learning of the students involved were analyzed. The results show that the automatic challenges generation is feasible, and the use of competitions in the teaching of Computer Security is motivating and effective for didactic purposes.**

***Keywords*—Automatic Problem Generation. Computer Security. Teaching.**

1. Introdução

Segurança Computacional é uma área cada vez mais importante e necessária, considerando a onipresença tecnológica e a necessidade de proteção das informações. Esta proteção envolve não apenas tecnologias, mas também conhecimento e adoção de boas práticas de Segurança por parte daqueles que desenvolvem, gerenciam e usam os sistemas computacionais que manipulam as informações [1].

Pesquisas têm mostrado que construir e sustentar uma cultura de Segurança é importante em tempos de mudanças radicais [2]. Contudo, muitas vezes a necessidade de difundir a Cibersegurança é negligenciada pelas instituições. Além disso, indivíduos capacitados em Segurança são necessários, mas o mercado carece deste tipo de profissional [3], havendo grande demanda por formação na área [4].

A educação formal, especialmente em nível de graduação, tem tido dificuldades para suprir essa carência. Um dos motivos é a defasagem dos currículos, que ainda precisam ser adaptados às diretrizes mais recentes. No Brasil, o Ministério da Educação instituiu diretrizes curriculares para os cursos de Computação apenas em 2016, considerando Segurança um dos temas que devem estar presentes entre as habilidades e competências para a formação profissional [5]. O relatório para diretrizes curriculares para cursos de TI, elaborado em 2017 pela ACM (*Association for Computing Machinery*) e pela IEEE-CS (*Institute of Electrical and Electronics Engineers Computer Society*), sugere que a área de Cibersegurança deve emergir nos novos currículos [6]. A realidade atual é que as poucas disciplinas que trabalham Segurança Computacional têm suas ementas elaboradas para abordar muitos assuntos, de forma genérica e conceitual, ou englobar poucos tópicos, mas com profundidade razoável. Muitas são ministradas de forma tradicional, usando livros e aulas expositivas, com enfoque teórico [7]. Percebe-se assim a necessidade de intensificar a formação em Segurança, enfatizando não apenas a teoria mas também a prática, e despertar o interesse dos alunos pelo tema.

Para atrair alunos e futuros profissionais para a área de Segurança Computacional, é importante ensinar por meio de metodologias que incentivem o discente. Uma forma motivadora para a introduzir os conceitos de Cibersegurança é através de jogos. Devido à motivação extra proporcionada por um ambiente competitivo, as competições de Segurança têm se tornado cada vez mais populares [8].

Um dos tipos mais usuais de jogos de Segurança Computacional é o *desafio* (ou *caça ao tesouro*), que consiste em encontrar palavras secretas (*flags*) ocultas em arquivos usando técnicas e ferramentas computacionais. Nos desafios, os problemas costumam ser criados manualmente, o que impõe algumas dificuldades. A criação de problemas é uma tarefa trabalhosa e exige conhecimento técnico especializado, o que limita a popularização desse tipo de jogo pela escassez de recursos humanos disponíveis. Após a sua realização, é comum que desafios sejam publicados na Internet, não raro com as soluções para os problemas, invia-

bilizando o reaproveitamento de questões, pois parte da dificuldade do jogo está em encontrar a estratégia para resolver cada problema. Além disso, problemas criados manualmente costumam ter uma *flag* única, igual para todos os jogadores. Como as *flags* são usadas como comprovação da resolução dos problemas, essa unicidade permite que elas sejam copiadas e/ou compartilhadas entre jogadores, possibilitando que estes tenham acertos computados mesmo sem efetivamente resolver os problemas correspondentes. Por fim, observa-se ainda que as dificuldades citadas reforçam-se mutuamente: à complexidade da criação de problemas somam-se a perda de valor após sua publicação e o compartilhamento de *flags*.

Visando a reduzir as dificuldades citadas e a facilitar a criação e a promoção de desafios, este trabalho propõe a geração e composição automática de problemas para esse tipo de jogo. A ideia é automatizar a geração de problemas a partir de um rol de técnicas que podem ser aplicadas de forma individual ou composta, e gerar competições inteiras formadas por conjuntos equivalentes de problemas, mas com *flags* únicas para cada jogador. A geração automática de problemas já foi abordada em outros trabalhos [9], [10]. As contribuições deste trabalho estão no uso mais extenso da composição de técnicas e na geração automática de competições, não apenas de problemas individuais.

O restante do artigo está organizado da seguinte maneira: a Seção 2 apresenta uma revisão geral de jogos usados no ensino-aprendizagem de Segurança. A Seção 3 discute os principais trabalhos relacionados. A Seção 4 detalha a geração automática de problemas para Segurança. A Seção 5 mostra a avaliação realizada e os resultados obtidos, e a Seção 6 apresenta as conclusões do artigo.

2. Jogos para Segurança

Inserir jogos para aumentar a consciência em Cibersegurança é uma prática que vem ganhando espaço [11], [12]. No que se refere ao valor pedagógico, não é nas competições que a maioria das habilidades são adquiridas [8], mas no período de preparação, através de treinamentos, estudos e troca de experiências. Há indícios de incremento nas habilidades técnicas de participantes de jogos descritos em diversos trabalhos [13], [14], [15], mas as vantagens destas atividades envolvem ainda outros aspectos, tais como liderança, trabalho em equipe (para jogos não individuais), tomada de decisão e controle de tempo [16].

Diferentes tipos de jogos são usados no ensino de Segurança, incluindo

- *videogames* [17], [18];
- jogos de tabuleiro e cartas [19], [20];
- competições de ataque e defesa [8], [11], [14]; e
- desafios (caça ao tesouro) [12].

Competições do tipo *desafio*, também chamadas de *caça ao tesouro*, consistem em conjuntos de problemas que precisam ser resolvidos com processos e ferramentas, tipicamente sem interação com outros jogadores. Elas motivam os jogadores a encontrarem recursos secretos (*flags*) e, ao mesmo tempo, consistem em atividades que podem ser facilmente reproduzidas, pois exigem um número limitado de ferramentas que possibilitem sua solução, tais como analisadores de tráfego de rede e editores de arquivos binários.

Os problemas de competições do tipo caça ao tesouro não preveem a interação direta entre jogadores/equipes, mas promovem a competição com base em sistemas de pontuação. Por exemplo, jogadores/equipes que cumprirem primeiro uma tarefa recebem mais pontos do que aqueles que terminarem depois, seguindo uma escala decrescente [8], ou pelo tempo de realização da tarefa, fazendo com que todos que cumprirem a tarefa até o tempo x recebam y pontos. Estabelecer alvos fixos para todas as equipes promove a competição de uma maneira mais saudável do que ocorre em jogos nos quais equipes atacam umas às outras [12]. Além disso, desafios são flexíveis a ponto de possibilitarem também a prática individual.

Desafios podem ser lineares, não lineares ou mistos. Em um desafio linear, os problemas precisam ser resolvidos em sequência, e geralmente têm dificuldade progressiva [21]. Em um desafio não linear, os problemas são independentes, e podem ser resolvidos em qualquer ordem. Nesse tipo de desafio, os problemas podem ter pontuações diferenciadas em função de sua dificuldade [22]. O formato ainda pode ser misto, agrupando problemas em fases sequenciais (portanto, linear), com cada fase tendo um conjunto de problemas que podem ser resolvidos em qualquer ordem [23].

Desafios vêm sendo aplicados há alguns anos fora do contexto acadêmico. O Departamento de Defesa dos EUA promoveu entre 2006 e 2013 o DC3 *Digital Forensics Challenge* [24], um desafio aberto a participantes de todo o mundo. Experiências no Brasil incluem o Cryptorace[1], desde 2015, e o Hackaflag[2], da Roadsec, e o Workshop de Forense Computacional do SBSeg, em 2015 e 2016. O público-alvo desses desafios, porém, não é bem definido, podendo misturar estudantes e profissionais.

Comparando os tipos de jogos, *videogames* e jogos de tabuleiro e cartas são geralmente mais preocupados em conscientizar os jogadores através de conceitos do que com as tecnologias atuais. Jogos de ataque e defesa exigem conhecimentos práticos mais aprofundados, tais como desenvolver códigos para explorar vulnerabilidades e corrigir serviços vulneráveis, e sua realização envolve uma logística complexa, incluindo infraestrutura computacional sofisticada e pessoal altamente especializado para criar um jogo e manter a infraestrutura [8], [12], [25]. Desafios exigem conhecimentos práticos com menor profundidade do que ataque e defesa, e são mais flexíveis na medida em que podem envolver problemas com variados níveis de complexidade. Desta forma, competições do tipo desafio podem propiciar competitividade aos jogadores mais preparados e, ao mesmo tempo, possibilitar que os iniciantes façam progressos e mantenham-se motivados.

O ineditismo é um fator motivador quando presente nos jogos, mas também traz uma dificuldade aos criadores. Muitas vezes as soluções podem ser encontradas após a

1. http://roadsec.com.br/cryptorace/
2. https://roadsec.com.br/hackaflag/

aplicação do jogo, tornando este praticamente descartável por perder o "fator surpresa". Este problema é chamado de *compartilhamento de problemas* [25]. Além disso, durante a aplicação do jogo pode ocorrer a cópia de *flags* encontradas por outras equipes, um problema chamado de *compartilhamento de flags* [9]. Esse problema ocorre porque nem todos os participantes têm interesse em ganhar, mas, por vezes, apenas em avançar etapas. Uma forma de contornar essas dificuldades é automatizar a geração dos problemas que compõem um desafio, usando de aleatorização para fazer com que os problemas gerados sejam únicos.

3. Trabalhos Relacionados

Esta seção examina trabalhos que utilizam geração automática de problemas (*Automatic Problem Generation* – APG) em desafios de Segurança.

O PicoCTF [9], [23] é um desafio voltado a estudantes, aplicado anualmente. Desde 2014 ele usa APG em problemas individuais para mitigar a ocorrência de cópias de *flags*, tendo sido identificado como o trabalho pioneiro nessa área. O MetaCTF [10] é um jogo de desafio que incorporou APG a partir de 2015 com o objetivo de mitigar as chances de trapaça. A competição foi voltada ao ensino de Engenharia Reversa e análise de *malware*, exigindo conhecimentos em ferramentas como `objdump`, `readelf` e `gdb`. O trabalho avaliou a qualidade e a utilidade de tarefas extraclasse pré-competição, tendo como resultado um aumento significativo no desempenho dos alunos nessas tarefas quando o jogo foi aplicado.

O SecGen [26] é uma ferramenta capaz de gerar desafios randômicos em conjuntos de máquinas virtuais (MVs), chamados de *cenários ricos*. O trabalho tem grande diversidade de problemas, tais como serviços de redes, esteganografia e vulnerabilidades em sistemas. Em uma competição, cada jogador recebe uma MV com um sistema operacional diferente e conjuntos distintos de problemas, o que minimiza o compartilhamento de técnicas e *flags*. As análises do trabalho indicam satisfação na interação com a ferramenta e adequação no nível de dificuldade dos problemas.

Todos esses trabalhos fornecem o jogo com sistema de placar, aplicam as atividades de forma não linear e permitem a criação de instâncias exclusivas. No entanto, embora o PicoCTF utilize APG, seus problemas não compõem mais de uma técnica. O MetaCTF utilizou APG em uma abordagem restrita a uma classe de problemas. Além disso, o trabalho relata o uso de codificação em um de seus problemas, mas não trabalha o conceito de composição de técnicas, ficando restrito a uma técnica e, eventualmente, adicionando uma forma de codificação ao problema. O SecGen permite aninhar técnicas somente compondo um problema com alguma forma de codificação, aplicada diretamente na *flag*. Nossa proposta aplica a composição de forma mais ampla, evitando apenas combinações inviáveis de técnicas. Além disso, as instâncias de problemas têm a mesma complexidade, enquanto o SecGen e o MetaCTF permitem que os jogadores recebam problemas totalmente diferentes, influenciando no placar e no esforço necessário para resolver o exercício.

4. Geração Automática de Desafios

Competições de Segurança costumam apenas atestar o conhecimento dos jogadores, trazendo as ideias de concorrência e superação. Em outras palavras, o público-alvo costuma ser formado por indivíduos com conhecimentos no assunto e que apenas buscam vencer a competição [27]. No âmbito educacional, competir não é o objetivo principal, mas um meio de influenciar positivamente os processos de ensino e aprendizagem, considerando que o jogador é também um estudante. Tão importante quanto reconhecer os jogadores mais capacitados é estimular os demais a aprofundarem seus conhecimentos em Segurança e a seguir carreira na área.

As competições podem ser individuais ou por equipes. A proposta deste trabalho considera o primeiro caso, para permitir individualizar a avaliação dos jogadores. No entanto, é possível jogar em equipes, a critério do organizador. A competição proposta é não linear, isto é, os problemas podem ser resolvidos em qualquer ordem. Para manter o placar, considera-se que todos os problemas têm peso 1, assim, conta-se apenas o número de acertos. Para jogadores com o mesmo número de acertos, o horário da última submissão correta é usado como critério de desempate.

Para a construção do protótipo da ferramenta geradora de desafios, foram analisadas 84 competições realizadas entre 2016 e 2017 descritas um repositório de desafios mantido pela comunidade [28], totalizando 1250 problemas que cobrem cerca de 200 técnicas distintas. Foram identificadas as 40 técnicas mais frequentes e, destas, oito foram escolhidas. Os critérios de seleção de técnicas foram o objetivo educacional do trabalho e o perfil do público-alvo, que seriam estudantes sem conhecimentos avançados em Linux e Segurança Computacional.

As técnicas implementadas foram:

- **Comentário em código-fonte de página HTML**: problema em que a *flag* é inserida arbitrariamente em um arquivo HTML válido. A página carrega estilos e imagens diferentes em cada instância.
- **Comentário no arquivo `robots.txt`**: problema que contém um conjunto de arquivos de um *website* e, entre eles, um arquivo `robots.txt` que contém uma quantidade parametrizável de comentários. Todos os comentários são sequências aleatórias de caracteres, e um corresponde à *flag*.
- **(De)codificação de arquivo em `base64`**: problema em que a *flag* é inserida em um arquivo de texto sorteado arbitrariamente em um diretório parametrizável. Após isso o arquivo é codificado em formato `base64`.
- **(Des)criptografia de Cifra de César**: problema em que a *flag* é inserida um arquivo de texto sorteado aleatoriamente em um diretório parametrizável. Após isso o arquivo é cifrado com a Cifra de César, com chave aleatoriamente escolhida entre 1 e 25.
- **(De)codificação de caractere ASCII para inteiro**: problema em que a *flag* é inserida em um arquivo de texto sorteado arbitrariamente em um diretório

parametrizável. Após isso o arquivo é codificado em valores inteiros, que representam os caracteres originais com base na tabela ASCII (*American Standard Code for Information Interchange*).

- **Descompilar binário e obter fonte Java**: problema em que um código Java, que produz um texto arbitrário, é criado. Neste código é inserida uma *flag* em uma variável e o código é compilado para *bytecode*, gerando um arquivo `.class`.
- **Descompilar binário e obter fonte Python**: semelhante ao problema anterior, mas usando Python em vez de Java e gerando um arquivo `.pyc` em vez de `.class`.
- **Esteganografia em imagens**: problema em que a *flag* é esteganografada em uma imagem escolhida arbitrariamente em um diretório parametrizável.

A Figura 1 apresenta duas instâncias do problema "Comentário em código-fonte de página HTML", geradas pela ferramenta, e seus respectivos códigos-fonte. Nota-se que as *flags* estão em linhas diferentes (linha 24 na imagem da esquerda e linha 27 na da direita) e têm sequências distintas de símbolos, embora sejam de mesmo tamanho. As imagens exibidas e o estilo das páginas são aleatórios, assim cada usuário recebe um arquivo com configurações diferentes.

Como as *flags* são distintas, o compartilhamento de respostas não produz efeitos. Por outro lado, a mesma estratégia pode ser usada para se chegar à resposta. O eventual compartilhamento de procedimentos de resolução é positivo, pois permite que os jogadores progridam no jogo e aprendam novas técnicas e ferramentas com seus pares.

O gerador de desafios também compõe técnicas, gerando instâncias distintas para cada jogador. A composição de técnicas se configura em aplicar uma técnica e depois submeter a *flag* ou o(s) arquivo(s) de saída à aplicação de uma segunda técnica – que pode ser a mesma novamente – para, então, gerar novo(s) arquivo(s) de saída. Por exemplo, um problema com a composição entre `base64` e César irá aplicar a codificação `base64` em um arquivo, gerando um arquivo de saída intermediário. Neste arquivo será aplicada a Cifra de César, gerando um novo arquivo de saída que compõe as duas técnicas em um problema, sendo este disponibilizado ao jogador. Assim, para resolver o exercício, o jogador precisa encontrar a chave utilizada na Cifra de César, obter o arquivo codificado em `base64`, decodificá-lo e procurar pela *flag*.

Cabe destacar que nem todas as técnicas podem ser compostas. Por exemplo, a composição (*Cifra de César* $\circ$ *Cifra de César*) não é viável, já que a aplicação da cifra várias vezes terá sempre uma chave equivalente à aplicação da cifra uma única vez. O gerador de desafios implementado produz apenas composições viáveis.

É importante citar que a ordem de aplicação das técnicas influencia na instância gerada. Por exemplo, aplicar codificação `base64` antes de um problema de descompilação de código implica gerar uma *flag*, codificá-la e depois inseri-la no código fonte. Caso a ordem seja invertida, uma *flag* é inserida no código fonte e, após a compilação, o arquivo de *bytecode* é codificado em `base64`.

A Figura 2 mostra uma instância do problema que compõe *Esteganografia em imagens* e *(De)codificação de arquivo em `base64`*. Nele o jogador recebe um arquivo codificado em `base64` e, ao decodificá-lo, obtém uma imagem que contém um texto esteganografado com a ferramenta `outguess`. Ao extrair o conteúdo da imagem com o `outguess`, obtém-se a *flag*. As *flags* geradas e as imagens usadas em cada instância são diferentes, o que produz arquivos diferentes quando codificados em `base64`.

A interação com a ferramenta geradora de problemas é feita pelo organizador da atividade, informando os parâmetros e criando os problemas desejados, que são instalados em um servidor *web*. O organizador atribui identificadores (IDs) aos jogadores, que usam um sistema *web* para baixar seu conjunto de problemas e para realizar a submissão das *flags* encontradas. O sistema *web* ainda fornece instruções básicas e placares individual e geral para acompanhamento da competição.

5. Avaliação

Para avaliar a eficácia da proposta em gerar desafios distintos sem trivializar os problemas e a percepção de satisfação, aprendizagem e interesse dos estudantes, foi realizado um experimento, conforme descrito a seguir.

5.1. Descrição da Atividade

A proposta do experimento foi realizar a competição em ambientes acadêmicos. O organizador usou a ferramenta desenvolvida para gerar os desafios. O experimento foi realizado em três etapas, sendo uma aula preparatória e duas competições. As competições foram individuais, com duração de 1h30min, realizadas em laboratório de informática com máquinas previamente configuradas com as ferramentas necessárias.

Na aula preparatória[3] foi explicada aos alunos a dinâmica do desafio e foi apresentado um conjunto de ferramentas que poderiam ser usadas na resolução dos problemas, com exemplos e exercícios. No início da aula os alunos assinaram um Termo de Consentimento sobre sua participação no experimento, e responderam a dois questionários, um de levantamento de perfil[4] e outro pré- teste[5], identificando suas impressões antes da competição.

Na aula seguinte, ocorreu a Competição 1 (`C1`). Foi explicado o funcionamento do desafio, foram distribuídos os IDs dos jogadores e o endereço de acesso ao servidor *web*, para então ser iniciada a competição. Todos os jogadores receberam instâncias de problemas com as mesmas técnicas. A escolha dos problemas se deu a partir das possíveis composições, de forma a utilizar todas as técnicas para as quais houve implementação no gerador de desafios. Seis

3. Arquivos disponíveis no repositório do trabalho: https://github.com/TreasureHuntGame/TreasureHunt.

4. Disponível em: http://bit.ly/2AdEWMW

5. Disponível em: http://bit.ly/2AoVkew

Figura 1. Duas instâncias do problema “Comentário em código-fonte de página HTML” e seus respectivos códigos-fonte.

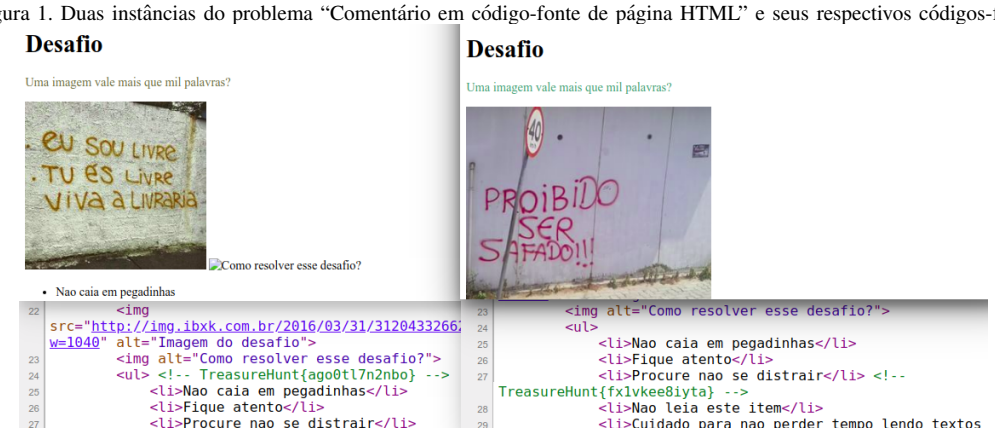
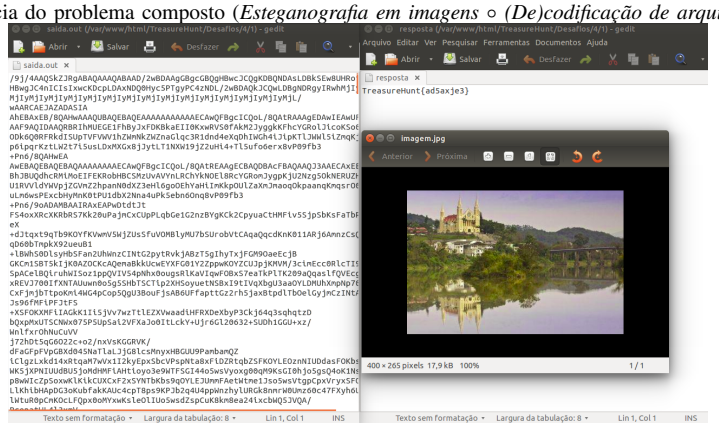


Figura 2. Instância do problema composto (*Esteganografia em imagens* ∘ *(De)codificação de arquivo em* `base64`).



problemas foram elaborados, sendo dois simples e quatro compostos. A escolha de exercícios simples e compostos foi feita para poder analisar a diferença de desempenho entre estes tipos. O tempo previsto para a atividade foi determinante para a decisão sobre a quantidade de exercícios, assim cada problema poderia ser resolvido, em média, em 15 min.

Na terceira aula, ocorreu a Competição 2 (`C2`). Novamente foi explicado o desafio, e foram distribuídos novos IDs aos jogadores. O desafio foi semelhante ao da Competição 1, porém, metade da turma resolveu os mesmos problemas (Grupo `C2.1`) constantes em `C1`, com a ordem diferente, e a outra metade (Grupo `C2.2`) resolveu problemas com as mesmas técnicas, mas com composições diferentes. A turma não foi avisada sobre a diferença dos exercícios. O critério de divisão dos grupos foi o desempenho em `C1`, de forma a mantê-los equilibrados. Assim, a colocação dos jogadores na Competição 1 e o número de acertos baseou a criação de dois grupos para os quais o somatório de acertos de seus jogadores na Competição 1 fosse igual ou o mais próximo possível disso. Em caso de quantidade ímpar de jogadores, o grupo C2.1 receberia um jogador a mais, em todas as turmas. O novo conjunto de exercícios foi escolhido pelos organizadores de forma a conter dois problemas individuais e quatro compostos, com complexidade semelhante. Ao final desta aula, os alunos responderam a um questionário pós-teste[6], registrando suas impressões após as duas competições.

As atividades foram realizadas com três turmas:

- **BCC**: turma formada majoritariamente por estudantes dos últimos semestres do curso de Bacharelado em Ciência da Computação da UDESC, na disciplina de Segurança de Redes de Computadores;
- **TADS**: turma do sexto semestre do curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal Catarinense – IFC; e
- **FIC**: turma de Segurança Computacional do Curso de Qualificação Profissional na área de Redes de Computadores do IFC.

Em todas as turmas o conjunto de exercícios das duas competições foi o mesmo. Participaram ao todo 30 jogadores, sendo 7 na turma FIC, 10 na turma TADS e 13 na turma BCC. As atividades foram realizadas pelos autores deste trabalho, na condição de organizadores, durante o mês de novembro de 2017.

5.2. Perfil dos Alunos

O questionário de perfil foi respondido por todos os 30 alunos. A média de idade foi de 23,0 anos, com pre-

6. Disponível em: http://bit.ly/2lYORDL

dominância de alunos do sexo masculino (93,3%) com formação em escolas públicas (76,7%).

Como indício de validação da adequação dos exercícios em relação a conhecimentos prévios e de motivação com a atividade, perguntou-se aos alunos, entre outras questões, sobre experiência com Linux e com Segurança Computacional e a motivação para cursar a disciplina. 43,33% afirmaram ter familiaridade moderada com Linux, com razoável conforto na linha de comando e criação de *scripts* simples. 36,67% responderam ser um pouco familiares, 13,33% levemente familiar e 6,67% extremamente familiares. Ninguém respondeu ser nada familiar. Vinte estudantes (66,67%), afirmaram não ter experiência com Cibersegurança. Seis (20,00%) afirmaram ter estudado por conta própria, quatro (13,3%) afirmaram estar fazendo a disciplina novamente e um (3,33%) fez um curso fora da instituição. Os resultados ratificam a premissa de que os jogadores não possuíam conhecimentos avançados, mas teriam condições de resolver as tarefas propostas.

No que diz respeito à motivação para cursar a disciplina, 56,67% responderam que achavam o assunto interessante, mas não tinham muito conhecimento, 6,67% pretendiam complementar a formação em Computação, 13,33% faziam para cumprir os requisitos do curso, 13,33% desejam seguir carreira em Segurança ou em uma área correlata e 10% forneceram outras respostas.

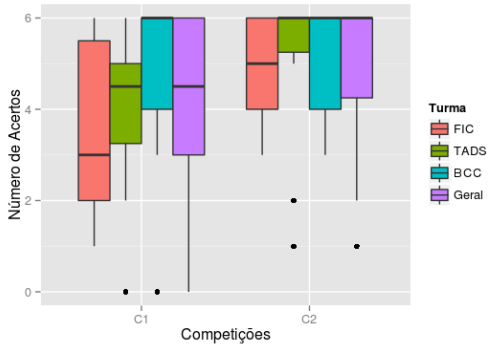
5.3. Resultados de Desempenho

Os resultados foram analisados com base em medidas de desempenho obtidas das respostas enviadas nas competições. Foram mensurados acertos e tempo de conclusão das atividades. Como as variáveis não eram normalmente distribuídas, para comparar o desempenho nas competições foram usados dois testes estatísticos não-paramétricos, o teste da soma dos postos de Wilcoxon (*Wilcoxon rank sum test*) e o teste de postos sinalizados de Wilcoxon (*Wilcoxon signed rank test*). O primeiro é um teste para amostras não pareadas, que compara grupos, e foi usado para comparar os dados de `C2.1` e `C2.2` na Competição 2, uma vez que não há interseção entre esses grupos. O segundo é um teste para amostras pareadas, que compara a evolução de indivíduos, e foi usado para comparar as competições `C1` e `C2`. Em todos os testes foi adotado o nível de significância $\alpha = 0{,}05$.

5.3.1. Número de Acertos. A Figura 3 apresenta o número de acertos das três turmas nas duas competições, por meio de *boxplots*. A figura mostra que a mediana na turma FIC ficou em 3,00 pontos, da turma TADS ficou em 4,50 pontos e da turma BCC ficou em 6,00 pontos. Em todas as turmas houve registros de estudantes obtendo a pontuação máxima, ao passo que houve registro de um jogador obtendo zero acertos em cada uma das turmas TADS e BCC. O resultado geral mostra que 75% dos alunos acertaram pelo menos a metade dos problemas, e que 25% acertaram todos. Esse desempenho permite concluir que a complexidade dos exercícios foi adequada para as turmas FIC e TADS. Na turma BCC a complexidade foi menos adequada, a julgar pela quantidade de jogadores (sete) que obtiveram o máximo de acertos já na primeira competição. A figura também mostra que, da `C1` para a `C2`, a mediana da turma FIC aumentou em 2,00 pontos e da turma TADS aumentou em 1,50 pontos, enquanto a da turma BCC manteve-se igual. Com base nos resultados gerais, percebe-se que a mediana ficou em 6,00 pontos. Ao todo, 60% (18 de 30) dos estudantes atingiram a pontuação máxima. A Figura 3 mostra que o desempenho dos estudantes melhorou e, para a maioria, foi fácil resolver os exercícios.

O teste de Wilcoxon para amostras pareadas identificou diferença estatisticamente significativa entre a quantidade de acertos nas Competições 1 e 2 ($V = 4{,}5$, $p = 0{,}00093 < 0{,}05$). Tem-se, portanto, que o desempenho dos jogadores na `C2` foi diferente, e superior, ao desempenho na `C1`.

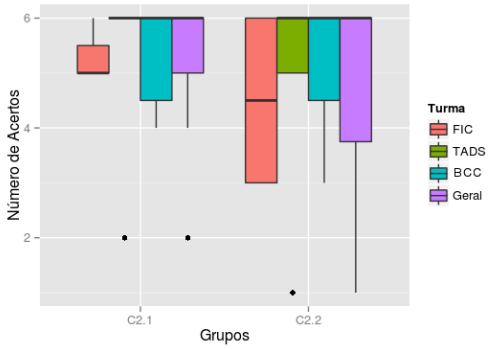
Figura 3. *Boxplots* de número de acertos nas competições 1 e 2.



Os resultados da Figura 4 mostram a comparação de acertos dos dois grupos da Competição 2 (`C2.1` e `C2.2`) em todas as turmas. Nota-se que tanto no grupo `C2.1` quanto no `C2.2` a maioria dos jogadores obteve a quantidade máxima de acertos, pois a mediana em todos os casos, com exceção da turma FIC, ficou em 6,00 pontos. O grupo `C2.2` apresentou maior variabilidade no número de acertos e uma quantidade maior de escores baixos, mas, de modo geral, o desempenho de todas as turmas foi considerado alto. O teste de Wilcoxon para amostras não pareadas em `C2.1` e `C2.2` não indicou diferença estatisticamente significativa ($W = 125{,}5$, $p = 0{,}54 > 0{,}05$). Portanto, a dificuldade diminuiu no segundo campeonato independentemente do grupo (`C2.1` ou `C2.2`).

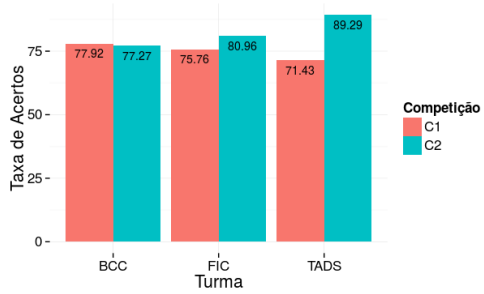
5.3.2. Taxa de Submissões Corretas. A taxa de submissões corretas por turma foi observada através de todas as submissões realizadas no sistema, e calculada dividindo o total de acertos pelo total de submissões realizadas. Esse índice pode ser diferente do apresentado na Seção 5.3.1 porque os jogadores podem efetuar várias tentativas até chegar à resposta correta, o que pode revelar problemas de comunicação, na transmissão das regras do jogo, no sistema ou tentativas de trapaça (compartilhamento de *flag* ou força bruta). A Figura 5 mostra a taxa de submissões corretas (em %) em todas as turmas, nas duas competições.

Figura 4. Comparação de acertos entre os grupos `C2.1` e `C2.2` em todas as turmas.



A média geral na Competição 1 foi de 75,25%, e na Competição 2 foi de 82,14%. Da Figura 5 é possível perceber que, com exceção da turma BCC, a taxa de submissões corretas foi maior na segunda competição. Parte dos erros ocorreu por inserção de espaços em branco nas *flags*, e foi possível perceber que nenhum estudante enviou *flags* já enviadas por outros colegas, ou seja, não houve compartilhamento de *flags*.

Figura 5. Taxa de submissões corretas (em %) das três turmas nas duas competições.

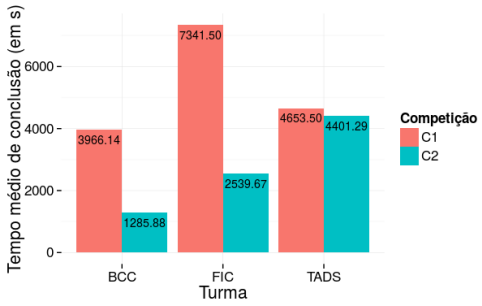


5.3.3. Tempo Médio de Conclusão. A Figura 6 mostra o tempo médio para conclusão da atividade entre os alunos que conseguiram obter 100% dos acertos. Para isso, foi considerado o horário de início da competição e o horário da última submissão correta de cada jogador. Nas turmas TADS e FIC, dois alunos gabaritaram a Competição 1; na turma BCC, sete. Na Competição 2, sete alunos gabaritaram na turma TADS, três na turma FIC e oito na BCC.

Percebe-se pela Figura 6 que o tempo médio para concluir a atividade baixou, e a quantidade de alunos que conseguiu acertar 100% dos exercícios aumentou nas três turmas. No entanto, é necessário ponderar que o tempo é afetado por vários fatores, tais como atraso dos estudantes e eventuais erros em ferramentas ou no servidor. O tempo médio geral da Competição 1 ficou em 3383,35 s (56,4 min), e na Competição 2 ficou em 2706,39 s (45,1 min), uma redução de 11,3 min em média.

O teste de Wilcoxon para amostras pareadas mostra que há diferença significativa entre os tempos de conclusão das Competições 1 e 2 ($V = 381$, $p = 0{,}0016 < 0{,}05$). Porém, o teste de Wilcoxon para amostras não pareadas mostra que a diferença entre os grupos `C2.1` e `C2.2` não é estatisticamente significativa ($W = 108$, $p = 0{,}89 > 0{,}05$). Este é mais um indício de que os exercícios ficaram mais fáceis após a aplicação da Competição 1, mas inconclusivos sobre a diferença de tempo para resolução de `C2.1` e `C2.2`.

Figura 6. Tempo médio para conclusão da atividade nas três turmas nas duas competições.



5.4. Resultados dos Questionários Pré e Pós-Teste

Os questionários pré e pós-teste ajudaram a avaliar o efeito da atividade e medir a percepção dos jogadores acerca de satisfação, aprendizagem e interesse sobre Segurança Computacional. O questionário pré-teste foi respondido por 30 alunos, e o pós-teste foi respondido por 29.

A análise dos questionários mensurou a receptividade dos alunos às competições e como elas contribuem para o aprendizado e influenciam sua perspectiva de carreira profissional. Também foi avaliada a existência de diferenças estatisticamente significativas entre as respostas dos questionários pré e pós-competição, o que evidenciaria uma mudança de percepção baseada na participação nos desafios. Para minimizar o viés nas respostas, optou-se por deixar os questionários anônimos (isto é, sem a identificação dos respondentes), o que inviabiliza a análise da evolução dos indivíduos. Assim, os grupos de respostas foram considerados como amostras independentes. Para essa análise foi usado o teste da soma dos postos de Wilcoxon (para amostras não pareadas), com nível de significância $\alpha = 0{,}05$.

Os aspectos de satisfação e percepção de dificuldade dos problemas foram objeto de múltiplas questões. Para medir a consistência interna das questões em mensurar cada um desses aspectos, foi efetuada uma análise de confiabilidade, usando o coeficiente alfa de Cronbach [29].

A comparação entre os questionários serviu para analisar possíveis mudanças nos resultados, podendo indicar se houve êxito em clarificar a ideia de competição pedagógica para o aluno, bem como aumentou sua motivação. A Tabela 1 exibe os resultados das questões e a Tabela 2 exibe as questões e seus respectivos identificadores.

Visando a identificar a satisfação dos jogadores com a atividade, observou-se, em especial, as questões 1.1, 1.2, 1.3, 1.5 e 1.7 do pré e do pós-teste. Essas questões usam

escala de Likert de cinco pontos, com respostas indo de *Discordo totalmente* (1) a *Concordo totalmente* (5). Valores superiores a 3 indicam satisfação positiva.

A Tabela 1 mostra que os resultados de satisfação ficaram acima de 4,00 pontos na coluna *Geral*, indicando que, na média, houve satisfação dos jogadores com a competição aplicada. A coluna *Evolução* contém a diferença entre as médias gerais do pós e do pré teste, e a coluna *Significativa?* indica se essa diferença é estatisticamente significativa ($p \leq 0{,}05$).

Percebe-se que todos os resultados de satisfação do pós-teste foram superiores aos do pré-teste. Houve evolução, em números absolutos, considerando os resultados por turma, com exceção da questão 1.3 na turma TADS, para a qual houve queda de 0,10 ponto, e da questão 1.5 na turma BCC, para a qual houve queda de 0,02 ponto. Há que se considerar também que nesta turma houve uma resposta a menos no questionário pós-teste em relação ao pré-teste.

Para mensurar a significância estatística das diferenças, aplicou-se o teste de Wilcoxon para amostras não pareadas, e constatou-se que para as questões 1.2, 1.3 e 1.5 a diferença encontrada não foi estatisticamente significativa ($p > 0{,}05$), mas o foi para as questões 1.1 e 1.7. É importante ressaltar que as médias para as questões de satisfação já eram altas (superiores a 4) no questionário pré-teste, o que limita a possibilidade de evolução, uma vez que a pontuação máxima possível é 5. Ainda assim, os resultados indicam que a participação nas competições aumentou a motivação associada a jogos como instrumento de aprendizagem e a percepção de que jogos podem despertar a atenção do público para Segurança Computacional.

A análise de confiabilidade referente às questões de satisfação teve um alfa de Cronbach de 0,78, que indica confiabilidade substancial, próxima do limiar para confiabilidade quase perfeita (0,8) [30]. Nessa análise foram consideradas apenas as respostas do questionário pós-desafio.

A questão 1.6 verificou se o aluno se sentia pronto para participar de desafios, medindo interesse e percepção do próprio conhecimento. As respostas estavam em escala de Likert de cinco pontos (*Discordo totalmente* ... *Concordo totalmente*). No geral, os alunos se sentiram pouco preparados para participar dessas competições. A média no pré-teste foi de 2,13, próxima da opção *Discordo parcialmente* na escala. A média no pós-teste aumentou para 2,79, mais próxima da opção *Neutro* na escala. Embora a percepção ainda seja mais negativa que positiva, houve uma evolução estatisticamente significativa de 0,66 pontos em relação ao pré-teste, indicando melhoria na percepção sobre o preparo com problemas de Segurança. Observando os resultados por turma, todas também apresentaram evolução.

A questão 2.1 investigou o interesse dos alunos em uma carreira em Segurança, abrangendo interesse e perspectiva profissional. As respostas estavam em escala de Likert de cinco pontos, indo de *Muito baixa* (1) a *Muito alta* (5). O resultado do pós-teste foi 3,00 pontos (neutro), 0,07 acima da média do pré-teste, não representando diferença estatisticamente significativa. Isso significa que os alunos não demonstram predisposição favorável ou contrária a uma carreira profissional na área de Segurança, e que a participação no desafio praticamente não alterou essa perspectiva.

Para identificar se o nível das questões foi adequado, utilizou-se como base as questões 3.1 e 3.2 do pós-teste. Os resultados, exibidos na Tabela 1, mostram que as médias para estas questões ficaram próximas de 3,00 pontos, indicando proximidade com a neutralidade. Essa neutralidade significa que os respondentes não acharam os problemas particularmente fáceis ou difíceis. A análise de confiabilidade referente às questões 3.1 e 3.2 resultou em um alfa de Cronbach de 0,72, o que indica confiabilidade substancial [30], ou seja, as respostas para as duas perguntas apresentaram concordância elevada.

As questões 4.1 e 4.2 do pós-teste versavam sobre a motivação dos alunos com a competitividade do jogo e a composição de problemas, respectivamente. As respostas estavam em escala de Likert de cinco pontos, de *Muito desmotivador* (1) a *Muito motivador* (5). Os resultados mostram que a competitividade foi considerada motivadora na média geral. Na turma BCC a média foi de 3,92 pontos. A média geral de 4,24 pontos na questão 4.2 indica que os alunos consideraram este fator entre *motivador* e *muito motivador*. Na turma FIC este resultado foi de 3,71 pontos. Os resultados inferiores a 4,00 pontos podem ter relação com a discordância de parte dos jogadores sobre a dificuldade dos problemas e o tempo gasto nas soluções.

A questão 1.4 contribuiu para a percepção de aprendizagem dos alunos. As respostas estavam em escala de Likert de cinco pontos, de *Discordo totalmente* (1) a *Concordo totalmente* (5). Ao contrário das outras questões, nesta questão as respostas positivas têm pontuações inferiores a 3, indicando que o aluno não tem dificuldade com as atividades. A média geral do pós-teste foi de 2,93 pontos, 0,24 pontos abaixo da média geral da mesma questão para o pré-teste (3,17 pontos). A diferença não é estatisticamente significativa, mas sugere que as competições tiveram um êxito moderado em diminuir a percepção da dificuldade com atividades práticas. O resultado também é condizente com os relatos sobre a discordância com a dificuldade dos problemas.

5.5. Observações

Durante as atividades na turma BCC, observou-se que alunos não matriculados na disciplina compareceram para acompanhar a atividade. Nesta turma, também foi possível verificar que o fator competição teve importância, com estudantes conversando sobre o que poderiam ter feito para obter resultados melhores ao final da primeira competição.

Nas turmas TADS e FIC não houve menção ao placar e à competição. Os alunos da turma TADS resolveram os problemas de maneiras diversas. Por exemplo, o problema de conversão de decimal para ASCII foi resolvido em serviços *on-line*, em substituição manual e por meio de codificação em JavaScript. O problema *Descompilar binário e obter fonte Python* foi resolvido com ferramentas de linha de comando Unix (`strings` e `cat`) e por ferramenta de descompilação *on-line*. Houve relatos de descontração durante as competições, tais como risadas ao concluir

Tabela 1. RESULTADOS DOS QUESTIONÁRIOS.

Questão	Atributo	BCC		TADS		FIC		Geral		Evolução	Significativa?
		pré	pós	pré	pós	pré	pós	pré	pós		
1.1	Satisfação	4,00	4,50	4,00	4,60	4,14	4,43	4,03	4,52	+0,49	Sim (p = 0,0076)
1.2	Satisfação	4,23	4,64	4,20	4,40	3,86	4,29	4,13	4,48	+0,35	Não (p = 0,12)
1.3	Satisfação	4,54	4,58	4,60	4,50	4,14	4,57	4,47	4,55	+0,08	Não (p = 0,51)
1.4	Aprendizagem	3,08	3,17	3,10	2,90	3,43	2,57	3,17	2,93	−0,24	Não (p = 0,43)
1.5	Satisfação	4,85	4,83	4,80	4,80	4,57	4,71	4,77	4,79	+0,02	Não (p = 0,81)
1.6	Interesse	2,15	3,05	2,20	2,40	2,00	2,86	2,13	2,79	+0,66	Sim (p = 0,018)
1.7	Satisfação	4,08	4,50	4,00	4,60	4,00	4,14	4,03	4,45	+0,42	Sim (p = 0,012)
2.1	Interesse	2,77	2,84	3,20	3,00	2,86	3,29	2,93	3,00	+0,07	Não (p = 0,75)
3.1	Dificuldade	–	2,85	–	3,10	–	2,71	–	2,90	–	–
3.2	Dificuldade	–	3,32	–	2,90	–	3,14	–	3,14	–	–
4.1	Motivação	–	3,92	–	4,00	–	4,14	–	4,00	–	–
4.2	Motivação	–	4,33	–	4,50	–	3,71	–	4,24	–	–

Tabela 2. ENUNCIADO DAS QUESTÕES ANALISADAS NA TABELA 1.

Questão	Enunciado
1.1	Jogos e competições me deixam mais motivado a aprender do que aulas expositivas
1.2	Eu gostaria que jogos e competições fossem explorados em outras disciplinas
1.3	Tenho interesse em atividades práticas envolvendo Segurança Computacional
1.4	Tenho dificuldade em atividades práticas envolvendo Segurança Computacional
1.5	Exercícios práticos de Segurança Computacional aumentam o entendimento sobre esta área
1.6	Sinto-me suficientemente preparado para (começar a) participar de competições de Segurança Computacional
1.7	Entendo que competições de Segurança Computacional aumentam o apelo desta área para o público geral.
2.1	A probabilidade de eu tentar seguir carreira na área de Segurança Computacional é
3.1	Os problemas do jogo aplicado foram difíceis de se resolver
3.2	Gastei muito tempo para resolver exercícios do jogo
4.1	Motivação com a competitividade do jogo
4.2	Motivação com a composição de problemas

exercícios ou ao decifrar códigos de pura distração, como no problema *Comentário em código-fonte de página HTML.*

Os benefícios da competição puderam ser ratificados com as avaliações aplicadas nas turmas FIC e BCC. Nestas turmas houve prova teórica, individual e sem consulta. Os estudantes da turma FIC lograram 100% de acertos em questões sobre técnicas vistas na competição, e os estudantes do BCC obtiveram, na média, 88,5% de acertos.

5.6. Discussão dos Resultados

De maneira geral, percebeu-se que os estudantes se interessaram pela área de Segurança Computacional. A solução de mais exercícios na Competição 2, e em menor tempo, bem como a evolução da percepção dos estudantes sobre o preparo para resolver problemas de Segurança, e da complexidade destes, foram indicativos de aprendizagem por parte dos alunos. Com base nos resultados, a percepção deles sobre motivação para uso de jogos em aula foi positiva.

Os resultados obtidos, tanto de desempenho quanto de questionários, indicam que os exercícios tornaram-se mais fáceis após a aplicação da Competição 1. A quantidade de técnicas selecionadas (oito) é um fator que pode ter contribuído neste sentido. Além disso, utilizou-se na Competição 2 as mesmas técnicas da Competição 1, mas com composições distintas para o grupo `C2.2`.

O número de acertos em problemas simples e compostos aumentou da `C1` para a `C2`, com diferença significativa estatisticamente. Além disso, a taxa de acertos para exercícios simples foi maior que a de exercícios compostos, o que já era esperado. Exercícios simples costumam exigir o uso de uma ferramenta para obter a solução, e são menos complexos que os exercícios compostos. Embora os exercícios deste trabalho sejam compostos em apenas dois níveis, a aplicação de uma técnica a mais traz maior complexidade para se solucionar de um problema. Este e outros resultados não foram detalhados por limitação de espaço.

A quantidade de jogadores com muitos acertos fez com que as medidas de tendência central apresentassem valores altos e as diferenças de desempenho não fossem estatisticamente significativas. Com isso, alguns resultados, tais como a diferença entre o desempenho dos estudantes dos grupos `C2.1` e `C2.2`, foram inconclusivos.

Observando o desempenho dos alunos e a análise das questões 3.1 e 3.2, nota-se um contraste: enquanto a maioria dos alunos obteve 100% de acertos na Competição 2, as respostas sobre a dificuldade dos problemas mostraram que, no geral, os estudantes não acharam os problemas fáceis ou difíceis, com resultado próximo do neutro neste item.

A geração automatizada de desafios de Segurança foi avaliada positivamente. Acredita-se que a eficácia de aleatorização de problemas pode ser melhor quantificada com desafios contendo mais problemas e com menor repetição de técnicas. As medidas que podem ser tomadas nesse sentido incluem incrementar a quantidade de técnicas e/ou a quantidade de opções em cada técnica e aumentar a quantidade de composições possíveis em cada problema.

6. Conclusão

O contexto atual mostra que ensinar Cibersegurança ainda é um desafio. Apesar do caráter contemporâneo e da importância relatada, o dinamismo inerente à área traz a necessidade constante de atualização. Além disso, é fato que são necessárias novas práticas pedagógicas para atingir melhores resultados para os estudantes. A adoção de jogos e competições são práticas que contribuem nesse sentido.

Este trabalho preconiza o uso de competições do tipo caça ao tesouro para o ensino de Segurança, e propõe o uso de aleatorização para gerar automaticamente instâncias diversificadas desse tipo de competição, contornando assim as principais dificuldades associadas. Em relação ao estado da arte, o trabalho traz como principais contribuições a possibilidade de gerar automaticamente competições inteiras, não apenas problemas individuais, e a utilização mais extensiva do conceito de composição de técnicas.

Competições produzidas com um protótipo do gerador de problemas foram realizadas em duas instituições e em três turmas. O efeito da competição foi avaliado através de questionários e do desempenho na atividade. A eficácia da ferramenta criada em gerar desafios distintos foi medida comparando grupos de alunos que receberam problemas com composições diferentes, mas não foi encontrada diferença estatisticamente significativa entre os grupos. Embora os resultados até o momento sejam encorajadores, a eficácia da aleatorização ainda precisa ser explorada mais a fundo. Com relação à percepção de satisfação, aprendizagem e interesse, avaliados através de questionários, os resultados forneceram indícios de que a atividade foi bem recebida pelos alunos.

Na continuação deste trabalho, deseja-se ampliar aplicar novamente os experimentos com amostras de tamanho maior e incluir novos problemas na ferramenta, com possibilidade de permitir que mais de duas técnicas componham um problema. Pretende-se ainda incluir identificadores às competições, o que permitirá observar a evolução de jogadores com o passar do tempo e isolar jogadores de turmas diferentes com mais facilidade para fins de análise de dados.

Agradecimentos

Os autores agradecem ao Instituto Federal Catarinense, por meio do Programa Institucional de Qualificação de servidores para o Instituto Federal Catarinense (PIQIFC), e à Universidade do Estado de Santa Catarina, que tornaram possível a realização deste trabalho.

Referências

[1] S. Furnell and N. Clarke, "Power to the people? the evolving recognition of human aspects of security," *Computers & Security*, 2012.

[2] G. Dhillon, R. Syed, and C. Pedron, "Interpreting information security culture: an organizational transformation case study," *Computers & Security*, vol. 56, pp. 63–69, 2016.

[3] R. S. Cheung *et al.*, "Challenge based learning in cybersecurity education," in *Proceedings of the 2011 International Conference on Security & Management*, vol. 1, 2011.

[4] M. Suby and F. Dickson, "The 2015 $(ISC)^2$ global information security workforce study," *Frost & Sullivan in partnership with Booz Allen Hamilton for ISC2*, 2015.

[5] MEC, "Resolução CNE/CES 5/2016," Diário Oficial da União, Brasília, 17/11/2016, Seção 1, p. 22–24, 2016.

[6] ACM, "Information technology curricula 2017," 2017, task Group on Information Technology Curricula. IEEE-CS. 2017 July 27.

[7] J. Mirkovic and P. Peterson, "Class capture-the-flag exercises." in *3GSE*, 2014.

[8] G. Vigna *et al.*, "Ten years of iCTF: The good, the bad, and the ugly," in *3GSE*, 2014.

[9] J. Burket *et al.*, "Automatic problem generation for capture-the-flag competitions," in *3GSE*, 2015.

[10] W. Feng, "A scaffolded, metamorphic CTF for reverse engineering," in *3GSE*, 2015.

[11] G. B. White and R. Dodge, "The national collegiate cyber defense competition," in *Proceedings of the Tenth Colloquium for Information Systems Security Education*, 2006.

[12] G. Vigna, "Teaching network security through live exercises," in *Security education and critical infrastructures*. Springer, 2003.

[13] R. Weiss, J. Mache, and E. Nilsen, "Top 10 hands-on cybersecurity exercises," *Journal of Computing Sciences in Colleges*, vol. 29, 2013.

[14] W. Petullo *et al.*, "The use of cyber-defense exercises in undergraduate computing education." in *ASE'16*, 2016.

[15] R. S. Cheung *et al.*, "Effectiveness of cybersecurity competitions," in *Proceedings of the International Conference on Security and Management (SAM)*, 2012.

[16] A. Conklin, "Cyber defense competitions and information security education: An active learning solution for a capstone course," in *HICSS'06*. IEEE, 2006.

[17] M. Guimaraes, H. Said, and R. Austin, "Using video games to teach security," in *ITiCSE*. ACM, 2011.

[18] M. Olano *et al.*, "SecurityEmpire: Development and evaluation of a digital game to promote cybersecurity education." in *3GSE*, 2014.

[19] T. Denning, A. Lerner, A. Shostack, and T. Kohno, "Control-Alt-Hack: The design and evaluation of a card game for computer security awareness and education," in *ACM CCS*, 2013.

[20] B. I. Gibson, "Educational games for teaching computer science," Master's thesis, Department of Computer Science and Software Engineering, University of Canterbury, New Zealand, 2013.

[21] E. Capuano, "#jolt hackathon 2017," 2017. [Online]. Available: https://blog.ecapuano.com/jolthackathon-2017/

[22] T. Chothia and C. Novakovic, "An offline capture the flag-style virtual machine and an assessment of its value for cybersecurity education," *3GSE*, 2015.

[23] P. Chapman, J. Burket, and D. Brumley, "PicoCTF: A game-based computer security competition for high school students." in *3GSE*, 2014.

[24] T. H. Lacey, G. L. Peterson, and R. F. Mills, "The enhancement of graduate digital forensics education via the dc3 digital forensics challenge," in *HICSS'09*. IEEE, 2009.

[25] C. Taylor *et al.*, "CTF: State-of-the-art and building the next generation," in *ASE'17*, 2017.

[26] Z. Schreuders *et al.*, "Security scenario generator (SecGen): A framework for generating randomly vulnerable rich-scenario VMs for learning computer security and hosting CTF events," in *ASE'17*, 2017.

[27] C. Eagle, "Computer security competitions: Expanding educational outcomes," *IEEE Security & Privacy*, vol. 11, no. 4, pp. 69–71, 2013.

[28] "CTF write-ups," 2017. [Online]. Available: https://github.com/ctfs

[29] A. Field, J. Miles, and Z. Field, *Discovering Statistics Using R*. SAGE Publications, 2012.

[30] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, 1977.