セキュリティ演習システムにおける試行錯誤機構の実行時間の短縮

竹原 一駿, 上 裕樹, 石塚 美伶, 亀井 仁志, 喜田 弘司, 最所 圭三

香川大学

s22d451@kagawa-u.ac.jp

Reducing Execution Time of Trial and Error Mechanism in Security Exercise System

Ichitoshi Takehara, Yuki Kami, Mirei Ishizuka, Hitoshi Kamei, Koji Kida, Keizo Saisho

Kagawa Univ.

概要

セキュリティ人材の育成手法として，サイバー防御演習の 1 つであるハードニング演習がある．我々は，演習中に演習状態を巻き戻すことで，何度でも防御方法をやり直せる試行錯誤機構を備えたハードニング演習システムを開発している．先行研究より，演習状態の巻き戻し (リストア) 後に，演習状態の保存 (セーブ) を行った際に処理時間が長くなることを発見し，その原因はディスク I/O にあると考えた．そこで，演習環境である仮想マシンを複数のディスクに分散して保存することで，処理時間を短縮できると考えた．本稿では，仮想マシンを複数のディスクに分散したことによるリストア後のセーブに関する評価について述べる．評価の結果，処理時間を短縮でき，従来よりも同時にセーブ/リストアを行える仮想マシンの台数が増加した．

1 はじめに

近年，世界的にセキュリティ人材の圧倒的な不足が報告されている [1]．セキュリティ人材を育成する手法として，サイバー防御演習の 1 つであるハードニング演習がある．ハードニング演習は，受講者がセキュリティ対応チームの一員になりきって，実際に運営されているサービスを模した演習システム上で，サービスを攻撃から防御する演習である．受講者は，サービスの運営を妨害する攻撃への対策を施すことで，セキュリティに関する知識と経験を得る．

従来のハードニング演習では，演習中に連続して演習シナリオに応じた新たな攻撃を受けるため，受講者は演習中に 1 つの攻撃に対して実行できる防御手法を 1 つしか試すことができない．また，チームワーク演習であるため，スキルの高い人が独走して演習を進めてしまう．

この問題を解決するために我々は，以下の特徴を持つハードニング演習を実施できるセキュリティ演習システム “ぷろてっくん” (Prote-kun) を開発している．

- 1 人で演習を遂行できる．
- セーブ/リストアにより，演習状態の保存と，任意の保存状態からの巻き戻しができる．
- 巻き戻した状態を同期することで，同じ攻撃の演習を何度でも試すことができる．

受講者は，演習の状態を巻き戻すことで，防御手法の良し悪しを比較しながら 1 人で学ぶことができ，攻撃に適する具体的な防御手法を学ぶことができる．

先行研究 [2] では，システムの開発と評価実験について述べた．評価実験の結果，演習状態のリストアを行った後のセーブは，非常に処理時間がかかることが判明した．我々は，その原因は，セーブ時のディスクへの書き込みにかかる時間が原因であると考えた．そこで，演習状態を保存する仮想マシンのイメージファイルを複数台のディスクに分散して保存することを考えた．これにより，ディスク I/O をディスク毎に並列に行えるようになり，処理時間が短縮できる．本稿では，ディスクを分散したことによる当該セーブにかかる処理時間の結果について述べる．

2 試行錯誤機構

試行錯誤機構は，演習を保存した状態から同じ攻撃を何度でも再現できる．この機構により受講者は，複数の防御手法を何度も試すことができる．

演習中の試行錯誤機構の使用イメージを，図 1 を用いて説明する． 演習開始直後はシステムによって自動

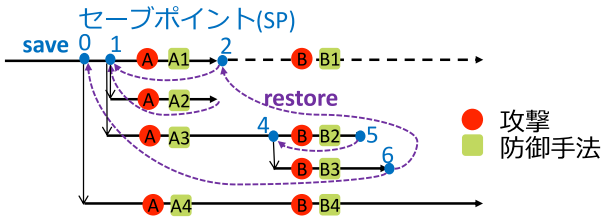


図 1　試行錯誤のイメージ

的に演習状態がセーブされ，セーブポイント (SP 0) が作成される．受講者は，予め演習状態をセーブし (SP 1)，攻撃 A に対し防御手法 A1 を実践する．その後，SP 2 として演習状態をセーブし，SP 1 にリストアし，再度攻撃 A を受け，今度は防御手法を変え A2 を実践する．同様に，セーブ・リストアを行い，防御手法 A3 などが行える．最初から演習をやり直したい場合は演習状態を SP 0 にリストアする．任意のセーブポイントにリストアすることができ，例えば，SP 6 としてセーブした後に SP 2 にリストアすることもできる．

3　システムの概要

“ぷろてっくん” は，演習を遂行するために，以下の仮想マシン (VM) を提供する．

- **防御用 VM:** 攻撃を受ける VM である．受講者は VM 上でサービスを運営し，防御手法を学習する．
- **攻撃用 VM:** シナリオに従って防御用 VM を攻撃する．演習シナリオには，攻撃のタイミングと内容が載せられている．
- **スコア計測用 VM:** 防御スコアを計測する．防御用 VM で動作するサービスの応答や防御度合いを確認する．

システム構成を図 2 に示す．各受講者に防御用 VM を割り当て，Web による演習画面を提供する． 攻撃用 VM とスコア計測用 VM は複数の防御用 VM で共有する．防御用 VM は，自身の演習時間を保持しており 1 分単位で増加させる．攻撃用 VM は，攻撃対象の演習時間に基づきシナリオに記載されている攻撃を防御用 VM に行う．防御スコアは，スコア計測用 VM が防御用 VM の演習時間に基づいて計測する．防御用 VM をリストアしたときにセーブしたときの演習時間も巻き戻ることになり， 攻撃用 VM は防御用 VM に対してセーブされた時間から同じ攻撃を行うことができる．セーブ/リストアは防御用 VM のみ行う．

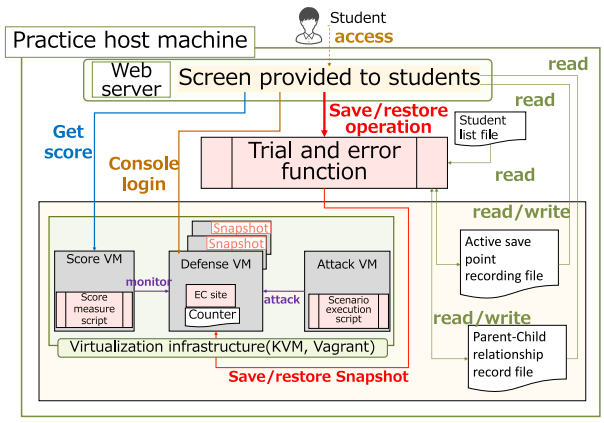


図 2　システム構成

仮想マシンのスナップショット機能を用いて作成した防御用 VM のスナップショットをセーブポイントとして用い，それらを管理するための試行錯誤機構を実現する．演習の進度や試行錯誤機能を用いるタイミングは受講者毎に異なるため，受講者毎にスナップショットを管理する．

4　先行研究における評価と課題

4.1　先行研究の評価

先行研究 [2] では，仮想マシンの構築とスナップショット機能を， 仮想化基盤 “KVM” と仮想マシン構築ソフトウェア “Vagrant” を用いて実装した．仮想ディスクのフォーマットは，スナップショット機能をサポートする “QCOW2” を用いた．

[2] では，表 1 に示すホストマシン上で，SSH と Apache をインストールした防御用 VM を 1 台のディスクに置き $(1, 2, 4, 8, 12, 16, 20, 32)$ 台に台数を変えながら，セーブ/リストアの処理時間を計測した．防御用 VM は，起動直後にセーブして “root” というセーブポイントを作成し防御用 VM に対して，試行錯誤機能を用いてセーブ/リストアと防御手法 (操作) を実行した．防御手法は，演習を再現する演習レシピを作成し実行した．演習レシピは，実際の演習より高い負荷となるように，短い間隔でセーブ/リストアを実行するように設定した．そのため，複数の防御用 VM に対して同時にセーブ/リストアが発生する確率が高い．時間計測には GNU の “`time`” コマンドを使用した．実験では，セーブ・リストアの時間を計測することが目的であるため，防御手法の適用時間の代わりに “`sleep`” コマンドを実行した．

実験の結果，リストアを行った後のセーブ (RS: Restore and Save) は，非常に処理時間が長くなることがわかった．図 3 に防御用 VM の台数毎の，RS の処理時間の平均と最大を示す． グラフから，受講者が試行錯誤機構を用いてストレスなく演習できるセー

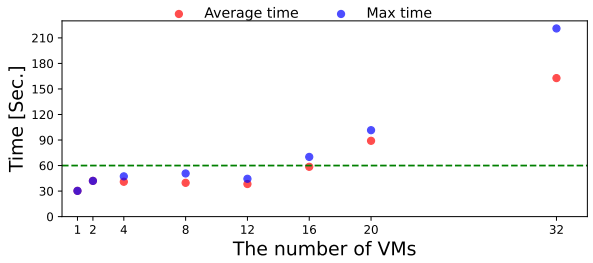


図 3　防御用 VM 毎の RS の処理時間の平均と最大 ([2] Fig.9 より参考)

ブ/リストアの処理時間を，[3] に従って 60 秒以内であることを要件とした場合，実験で用いたホストマシンで演習を実施できる受講者数が 12 人であると結論づけた．

このことから，RS の処理時間に注目することにした．

4.2　提案システムの課題

台数が少ないときには RS の処理時間は十分に短いことから，RS の処理時間はディスク I/O がボトルネックになっている．

そこで，機器に搭載されたディスクを物理的に増やし，防御用 VM の仮想ディスクを分散させることで，RS の処理時間を短縮できるか調べることにした．

5　評価実験

5.1　目的と手法

防御用 VM を格納するためのディスクを 3 台まで増やし，セーブ・リストアにかかる時間を測定し，演習可能な台数が増えることの確認を目的とする．実験手法は [2] と同じ手法を用いる．

本稿では，[2] で使用したマシンにディスクを増設できなかったため，先行研究より CPU の性能が高い表 2 に示すマシンで実験した．

実験手順を示す．まず，[2] と同様に 1 台のディスクで実験し，防御用 VM が 40 台，50 台の場合も追加して実験する．各処理時間のグラフの概形が [2] との差異が無いことを確認するとともに，当実験の機器を使用した場合の要件に従う仮想マシンの台数を求める．

表 1　[2] の実験で使用した機器のスペック

item	Value	
CPU	Intel Core i5-1145G7	4 Core 8 Threads
Memory	32GB	
NVMe	256GB	“/” and Swap Memory (32GB)
SSD	512GB	“/var” (Save VM images)
	512GB	“/home” (Executate this system)

次に，防御用 VM を 2 台と 3 台のディスクに分散させたときの RS の処理時間を計測することで，分散させたことの効果を求める．

5.2　評価結果

32 台の場合は，図 4 に示すようにメモリ・スワップが発生した．

図 5-(a) にディスクが 1 台のときの RS の処理時間の最大，平均を示す．緑の線は要件である 60 秒を示している．前述の通り CPU を変更したため，[2] の Fig. 9 (図 3) よりも全体的に処理時間は短くなっている．そのため，先行研究では要件を満たしていなかった 16 台，20 台も要件を満たしている．しかし，先行研究とグラフの概形は変わっていないことから，防御用 VM を複数のディスクに分散させる効果の検証には影響がないと判断した．

図 5-(b) にディスクが 2 台のとき，図 5-(c) にディスクが 3 台のときの RS の処理時間を示す．ディスクを増やすことで RS の処理時間を短縮できたことがわかる．特に，ディスクが 2 台のときは防御用 VM が 40 台まで，ディスクが 3 台のときは 50 台の場合でも，要件を満たすまでに短縮できた．

5.3　考察

ディスクを増やした場合の処理時間について考察する．

ディスク台数毎の処理時間の推移について述べる．“ぷろてっくん” で行う演習では，3 章より受講者 1 人あたりに防御用 VM を 1 台割り当てることとしている．要件に従いつつ試行錯誤機構を用いた演習は，

表 2　本稿の実験で使用した機器のスペック

item	Value	
CPU	Intel Core i5-12500	6 Core 12 Threads
Memory	32GB	
NVMe	256GB	“/” and Swap Memory (32GB)
SSD (sda)	512GB	“`/var`” (Save VM images)
	512GB	“`/home`” (Executate this system)
SSD (sdb)	1TB	“`/mnt/sdb1`” (Save VM images)
SSD (sdc)	512GB	“`/mnt/sdc1`” (Save VM images)

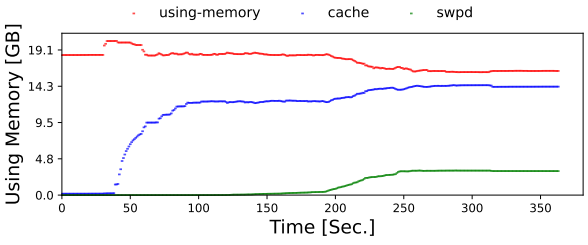
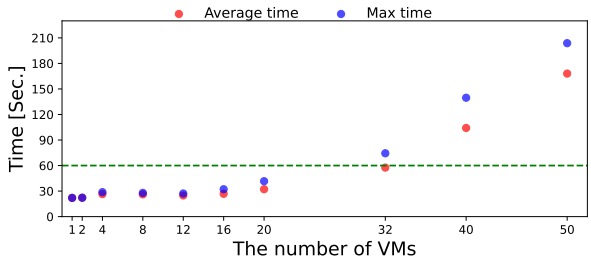
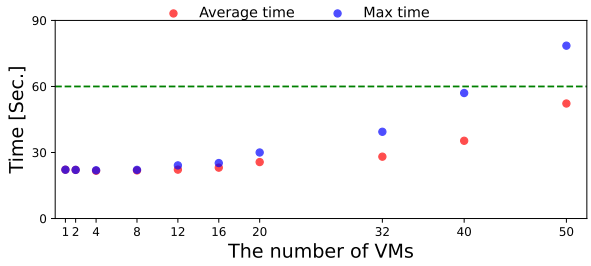


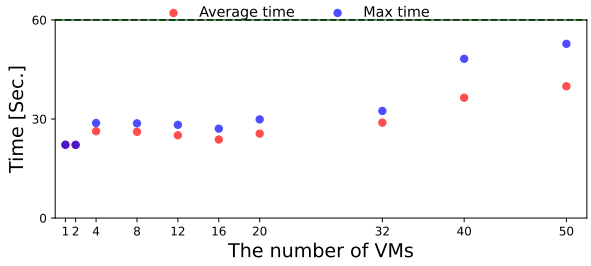
図 4　防御用 VM が 32 台のときのメモリ使用量



(a) 1 台の場合



(b) 2 台の場合



(c) 3 台の場合

図 5　ディスク台数毎の RS の処理時間の平均と最大

ディスクが 2 台ならば 40 人，3 台ならば 50 人まで可能である．著者らの所属する大学の 1 授業あたりの受講者数は，40〜50 人である．そのため，当実験と同様の構成の機器とディスクを 3 台以上用いることで，要件を満たしつつ演習が可能である．また当実験で使用した演習レシピは，高頻度にセーブ/リストアを実行する演習レシピを用いて行っており，同タイミングに行うセーブ/リストアの台数が頻繁に多くなる条件だった．実際の演習で同じタイミングで行う台数は少ないため，当実験結果は最も負荷がかかる条件として十分に有効であると考えられる．

また，32 台以上に着目すると前節で述べた通りスワップが発生しており，台数が増える毎に処理時間が長くなっていくため，ディスクの台数を増やした場合の RS への処理時間への影響を正確に測れていない．そこで，今後はメモリを増やして実験を進めることで，ディスク 1 台あたり何台の防御用 VM が要件を満たすことができるかを求める．

6　おわりに

本稿ではセキュリティ演習システムにおける，演習状態を巻き戻せる試行錯誤機構の処理時間の短縮について述べた．先行研究からディスクがボトルネックになると考え，仮想マシンを配置するディスクを分散することで，処理時間の短縮を実現した．これにより，処理時間に対しストレスなく演習可能な学生数を，先行研究の 4 倍以上に増やすことができた．

今後は，CPU，メモリ，ディスクの台数など，“ぷろてっくん” を構築する機器のスペックに応じた仮想マシンの台数を算出する式の確立や，更に高速なディスクを用いることで処理時間が長時間化する原因を調査する．

参考文献

[1] (ISC)^2. “Cybersecurity Professionals Stand Up to a Pandemic”, (ISC)^2 Cybersecurity Workforce Study, 2020.

[2] Ichitoshi Takehara, Mirei Ishizuka, Hitoshi Kamei, Koji Kida, Keizo Saisho. “Evaluation of Processing Time in Trial-and-Error Function of Security Exercise System for Security Beginners”, Proceedings of the 2023 World Congress in Computer Science, Computer Engineering, and Applied Computing, 6 頁, 2023 (印刷中).

[3] Jakob Nielsen. “Powers of 10: Time Scales in User Experience”, Nielsen Norman Group., https://www.nngroup.com/articles/powers-of-10-time-scales-in-ux/, 2023/09/21.