DF&OF～守るか攻めるかどっちなんだい？～

エントリー番号:12｜西の国会疑似堂

要旨

「DF&OF～守るか攻めるかどっちなんだい？～」というパスワードの重要性や攻撃側の手法を理解してもらうためのツールを提案する。大学入学共通テストで情報が必須となったことに加え、近年中高生のスマートフォンやタブレット端末の所持率が上がる中、このような機器を初めて扱う人でも理解しやすいように工夫している。具体的には対戦形式とし、パスワードを作成する利用者側の人とパスワードを破ろうとする攻撃者で**仕切る**、進行役がゲームを**指揮る**。そして、攻撃者はより早く特定できるように、パスワードを作成する人は破られにくいパスワードを作成するように知恵を出**し切る**ゲームとなっている。

1.背景

不正アクセスやパスワードの流出という言葉をよく耳にすることが、スマートフォンやタブレットを使用するユーザの増加と共に増えている。自分で管理するスマートフォンやタブレットを持つ時期の若年化が急激に進んでおり、中高生のスマートフォン所有は、当たり前になりつつある。しかし、適切なパスワードの取り扱いを学ぶ機会がほとんど無く、攻撃者にとって最高のターゲットになり得る。また、急激な IT の成長と浸透により、情報セキュリティを学んでいない親世代が子どもの安全の確保を出来ていないという現状もある。

大学入学共通テストに情報科目が採用されたことや、中学・高校での情報科目が重要視される中、子ども自身でパスワードの作り方や攻撃手法などの適切な知識をつけてもらうために今回の提案を行う。なお、攻撃手法に関しては、守るための手法を知らないといけないので、攻撃するために学ぶのでなく、守るために学ぶものである。

2.従来の取り組みの課題と本提案における解決方法

従来の取り組みとして、高校や中学校での授業が挙げられる。2025 年度から、大学入学共通テストでの情報科目の追加が決定したことにより、高校では、更に IT に関する教育が進んでいく。教科書や共通テストの難易度は、基本情報技術者試験相当であり、範囲は広いものの基礎的なものとなっている。IPA(独立行政法人 情報処理推進機構)の指標でも基本的技術・技能と情報処理技術者としては、最低限のものとなっている。

一括りに情報と言っても、ネットワークやセキュリティなど専門性が高い分野がいくつか集合しているので、全分野で正しい知識を教えることができる教員が少ないと考えられる。そのためセキュリティ分野では、パスワードの使いまわしは良くないと教えるぐらいで、具体的にどういうパスワードが危険なのか、どのようなことをすると危険なのかを詳しくは教えていないことがほとんどである。実際にどのようにすると危険なのか、実践しながら教える場が必要である。そこで、実際に被害を受ける想定の演習を行うことにより、危機感を持つことができると考えられる。

現時点でセキュリティに関する演習形式での取り組みの一つに、IPA が主催している「セキュリティキャンプ」というイベントがあり、その中の小中学生向けコースでセキュリティについての講義が開催されている。しかし、対象となる人物として、既にソフトウェアを開発した経験がある人、または興味のある人であるため、あまり知識が無い小中学生が応募するにはハードルが高くなっている。このことから、興味がある人以外に対して、セキュリティを演習形式で学ぶ機会が少ないと考えられる。高校生向けのイベントでは、さらにレベルが高くなって

いる。そのため、今回は中高生向けの初心者用演習教材を提案する。小学生や親世代でも十分に理解出来るようにも設計されている。

中高生にとって馴染みやすいものは、ゲーム仕立てである。パスワードを作成する人、対、攻撃する人の対戦形式で行う。決められた回数・時間内に破ることができれば攻撃者の勝ち、破ることができなかったら作成した人の勝ちという様に実践しながら対決し、楽しく学んでいく。

また本システムは、大人の科学から発売されている電子ブロックから着想を得た。電子ブロックとは、遊び感覚で電子回路を学ぶことができる製品である。トランジスタや抵抗、コンデンサといった汎用性の高いブロックを組み合わせて、大人から子どもまで、誰もが簡単に電子回路の仕組みを理解していくことができるという特徴がある。このメリットを本システムにも取り入れることでノーコード(実際にコードを書かずにプログラミングすること)を実現し、プログラミングや予備知識に関する敷居が徹底的に低くなるように実装した。

犯罪に巻き込まれるリスクの高い中高生に、ゲーム感覚でセキュリティに対する守備能力を養ってもらえると我々も嬉しいことこの上ない。

3.提案システムの使用方法

3.1 概要

本提案は、安全なパスワードの重要性を遊び感覚で身に付けることができる。概要を説明すると、文字当てのような遊びである。役割を攻守で「**仕切る**」とともに進行役がゲームを「**指揮る**」。ゲーム仕立てにすることで知恵を出「**し切る**」ことにより、より破られにくいパスワードを生み出すことと破る方法を考えることによって展開されていく。詳細は以下に記載する。

3.2 パスワード考案と攻撃の流れ

攻撃者(1 チーム or 複数チーム)と攻撃を受ける人(以下:被攻撃者、1 チーム)は、架空の人物のプロフィールを入手する。その情報を基に、被攻撃者は、PIN コードといくつかのパスワードを考える。ただし、被攻撃者は、架空の人物のプロフィールを見て考えていいが、作成したパスワードのメモなどを見ないで、パスワードを再現出来なくてはならない。図 3.1 は架空の人物のプロフィールカードの例である。

図 3.1 プロフィールカードの例

対戦は 3 回行い、攻守は変わらない。勝敗は、攻撃の実行時間が規定の時間より早く攻撃に成功したら攻撃者の勝ち、規定の時間以上かかった場合や、全くパスワードを解読できなかった場合は、被攻撃者の勝ちとなる。

1 回目の対戦では、被攻撃者は 4 桁の PIN を架空の人物の情報をもとに考える。攻撃者は、ブロックを並べ攻撃の手順を指定する。ブロックとしては、0、1、2、・・・9 をそのまま出力するものと、0~9 のいずれか 1 文字

を順番に出力するものを用いる。特定の数字の出力には▲秒、0~9 のいずれか 1 文字の出力 1 回につき〇〇秒かかる設定とする。▲秒や〇〇秒は、適宜設定する。PIN は 4 桁なので、ブロックは 4 つ並べることになる。4 つのブロックの配置は n 回まで指定できるものとし、トータル m 秒で特定出来なければ失敗となる。n 回や m 秒も適宜設定する。

図 3.1 の人物を例として説明する。攻撃者は、このプロフィールをもとに最適な攻撃手順を考える。この人物の誕生日は、2000 年 8 月 8 日である。PIN コードに 2000 や 0808 などの誕生日に関連するものを利用する可能性がある。仮に 2000 を指定した場合、1 単位秒×4 の 4 単位秒となる。ここで単位秒という単位を使っている。実行時間は、端末のスペックに依存するので、目安という意味で単位秒とする。逆に、4 桁と短い字数ということを逆手に 0000~9999 まで全ての数字を総当たりで攻撃する場合は、10000n 単位秒かかる。 これを n 回かつ m 秒以内で成功させることができれば、攻撃者の勝ちとなる。この際、攻撃者の考える時間は、特に指定がない。各自のレベルに合わせて設定してもいいだろう。

2 回目の対戦では、4 文字以上 8 文字未満の英字小文字と数字の組み合わせが可能な文字列でパスワードを作る。1 回目と同様に架空の人物情報をもとにパスワードを作成する。攻撃者は 1 回目と同じようにブロックを並べ攻撃の手順を指定する。ブロックとしては、0、1、2、・・・9 をそのまま出力するものと、0~9 のいずれか 1 文字を出力するもの、a、b、c、・・・z をそのまま出力するもの、a~z(アルファベット小文字)のいずれか 1 文字を出力するものを用いる。加えて、攻撃者は、プロフィールから推定される単語や”password”や”QWERTY”などの一般的に多用される単語が確認されているデータベースのいずれかを出力するものも選択肢として使用できる。特定の数字や文字の出力には▲秒、0~9、a~z のいずれか 1 文字の出力 1 回につき〇〇秒、プロフィールから推定される単語のデータベースから 1 つ出力するのにかかる時間を×秒と設定とする。ここでも×秒は適宜設定する。データベースのいずれかを出力するものを用いない場合、4 文字以上 8 文字未満なのでブロックは 4~8 個並べることになる。データベースのいずれかを出力するものを用いる場合、データベースに格納されている文字の長さにより並べる個数が変わる。図 3.2 は、4 文字で構成されているパスワードを解読する場合の並べ方の例である。ここで使用したデータベースというブロックは、4 文字のものを使用したとし、ブロックを一つ並べている。ブロックの配置は n 回まで指定できるものとし、トータル m 秒で特定出来なければ失敗となる。

1 回目と同様に例をあげる。アドレスの一部をパスワードに使用するケースも見られる。攻撃者が hm19secu をパスワードと予想した場合、すべての文字を総当たりで全ての文字列を調べる場合は、(36 文字種の 8 乗)単位秒かかる。ただし、この文字列がデータベースに格納されている可能性がある。8 文字のデータベースに 20 単語あった場合は、160 単位秒となる。オーダーとしては、80 単位(160/2)秒となる。データベースから正解を導き出したほうが、総当たりで文字列を導き出すより速い。これを n 回かつ m 秒以内で成功すれば、攻撃者の勝ちとなる。

図 3.2 四文字で構成されているパスワードを解読する際のブロックの並べ方の例

3 回目の対戦では、8 文字以上 16 文字未満の大文字小文字の区別有りの英字、数字、記号(,._!?@-)が使え、大文字英字、小文字英字、数字は必ず含まないといけないものとする。1,2 回目と同様に架空の人物情報をもとにパスワードを作成する。ブロックとしては、0、1、2、・・・9 をそのまま出力するものと、0~9 のいずれか 1 文字を出力するもの、a、b、c、・・・z をそのまま出力するもの、A、B、C、・・・Z をそのまま出力するもの、a~z(アルファベット小文字)のいずれか 1 文字を出力するもの、A~Z(アルファベット大文字)のいずれかを出力するもの、プロフィールから推定される単語のデータベース、記号(,._!?@-)のいずれかを出力するものを用いる。特定の数字や文字の出力には▲秒、0~9、a~z、A~Z のいずれか 1 文字の出力 1 回につき○○秒、プロフィールから推定される単語のデータベースから 1 つ出力するのにかかる時間を×秒、記号(,._!?@-)のいずれか 1 文字の出力 1 回につき□□秒かかると設定する。□□秒は適宜設定する。8 文字以上 16 文字未満なのでブロックは 8~16 個並べることになる。ブロックの配置は n 回まで指定できるものとし、トータル m 秒で特定出来なければ失敗となる。

攻撃の流れをまとめる、まず、攻撃者は攻撃の手法を考え、それに合わせてブロックを並べる。そして、カメラで読み取り、読み取った内容から攻撃のプログラムをアプリ内で自動生成する。自動生成したプログラムを用い、事前に考えた被攻撃者のパスワードに攻撃をしかける。攻撃の時間を計測し、勝敗を決める。

これを攻撃者と被攻撃者を入れ替えながら体験していくことにより、特定しやすいパスワードや PIN は、どのようなものなのか、ユーザと攻撃者の両方の立場になって考え、学ぶことが最終的な目標である。

4. 工夫したこと

4.1.ブロックの読み取りとコードの生成

画像処理には、OCR（Optical Character Recognition:光学式文字認識）という技術がある。これは画像データから文字を認識し文字データとして取り込む技術である。この技術を使用して並べているブロックに印字されている文字を読み取り(図 4.1)、ユーザー（攻撃者）が作成した攻撃手法をプログラムコードに変換する。

図 4.1.OCR での読み取り

攻撃者が横 1 列に並べたブロックを撮影し、OCR で攻撃の手順を読み込む。OCR で読み取った攻撃内容・量に合わせて図 4.2 のようにプログラムコードを生成する。図 4.3 はコードの処理をイメージしやすくした図である。

総当たり攻撃の場合は、特定するパスワードの桁数に合わせて A~Z のブロックを並べる。A~Z のブロックが一つの場合は一重の繰り返しループ、ブロックが二つの場合は二重の繰り返しループのプログラムコードを生成する。リスト攻撃の場合は、リストに載っている単語を読み込む繰り返しループとしてコードを自動生成する。コードの自動生成に関しては、カメラで読みとった際に専用アプリケーションが自動で行う。実行の詳細については、4 章 2 節で述べる。

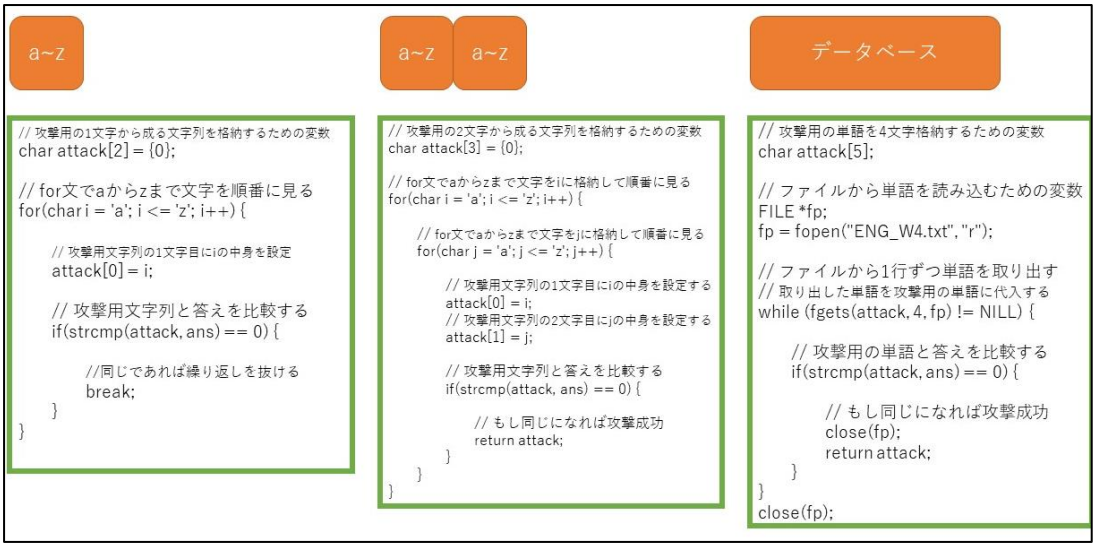

図 4.2.作成するコード例

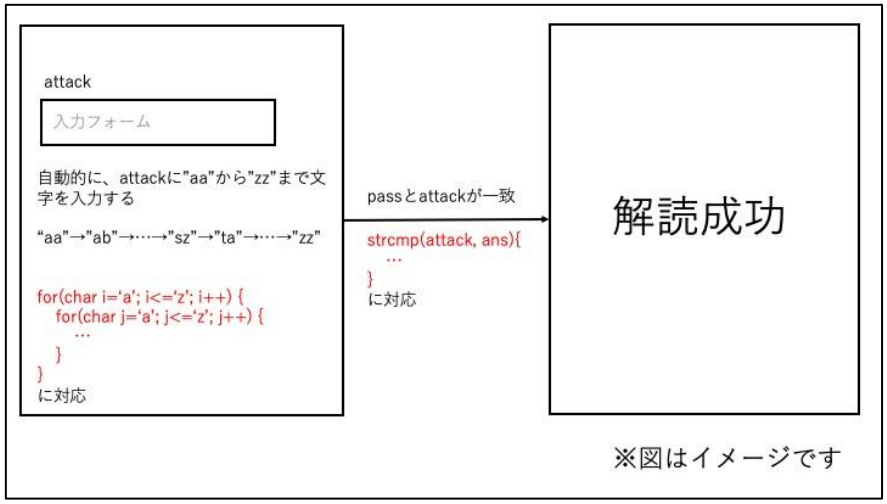

図 4.3.コード処理のイメージ図

4.2.容易に学校現場へ導入

演習を授業に導入する際に、ハードルとなる原因として、授業進行の難しさや、機材の導入の難しさという点が挙げられる。しかし、本提案で必要となる機材は、グループ毎のタブレット端末 1 台と専用アプリケーションのみである。さらに、グループ分けを行っているので、生徒全員分の端末を用意する必要が無い。そのため、簡単に演習を導入でき、設定の手間も省くことが出来る。さらに、授業のおおまかな進行手順の例も本節で示すため、スムーズな授業進行が可能である。

専用アプリケーションを用いた授業進行手順の例を、以下に示す。

① 生徒を攻撃グループと被攻撃グループに分け、タブレットを配布する。
② 被攻撃者に、仮想のプロフィールに沿ったパスワード作成を促す。
③ 専用アプリケーションに被攻撃者が考えたパスワードの入力を促す。
④ 攻撃者に、ブロックを用いて仮想のプロフィールに沿った攻撃手法の作成を指示する。
⑤ 攻撃者に、ブロックをタブレット端末のカメラで撮影して攻撃手法・内容の読み込みを促す。
⑥ 攻撃者に、攻撃ボタンを押すように指示し、読み込んだ内容に合わせて、実際に攻撃する。

⑦ タブレット端末に勝敗と解説が表示されるので、読むように指示する。

⑤のカメラでブロックが撮影される際に、自動的にコードが生成される。その状態で、専用アプリケーション画面内の攻撃ボタンを押すと、生成されたコードが実行される。

本提案の利点としては、先ほど挙げた、使用する機材が少なくて済む点と、演習を行う際に使用する専用アプリケーションの操作が容易であるという点がある。

4.3.計算コストの概念を直感的に体感

一般的にパスワード攻撃の手法毎に必要とする計算時間は異なることから、本提案においても攻撃に使うブロックごとに攻撃にかかる時間が計算コストとして設定されている。直感的に分かりやすく提示するために、ブロックごとに異なる重さを物理的に設定する。つまり、計算時間がかかるブロックは重く、さほど計算時間がかからないブロックは軽くなっている。

実際の攻撃では、莫大な時間がかかる上、それを試すことも難しいので、重さから実行時間(計算コスト)がわかるのは、初心者でもわかりやすい。わかりやすいという意味では、従来は PC などのデバイスを見て、直感的ではない操作で学習や攻撃の体験をしていたが、ブロックで簡単にプログラミングできるという点も挙げられる。

計算コストは勝敗に関係する。パスワードが破れても計算コストがかかりすぎた場合は攻撃者の敗戦となる。また、複数の攻撃者チームと単一の被攻撃者の対戦の場合は、パスワードを破るまでの時間コストが最も小さいチームが勝利となる。攻撃者と被攻撃者が一対一の場合はパスワードを時間内に破れたかどうかで勝敗が決まる。攻撃者が複数いる場合、パスワードを破るまでの時間と計算コストの総合判定によって勝敗が決まる。

5.むすび

この学習法が確立すると、若年層に対するセキュリティ教育を実践形式という形で授業に組み込むことが出来る。その結果、パスワードが解読される危機感を身近に感じることが可能になる。簡単な単語や推測されやすいパスワードを設定した場合の危険性を学ぶことが、セキュリティ被害の軽減にも繋がっていくだろう。また、演習中に使用するブロックの組み合わせから、アルゴリズムや計算量の概念が身につき、プログラミングをする際に効率の良いコードの書き方を身に付けることができる。

今後の課題として、より複雑なコードの記述の対応、ローコードレベルのプログラム記述環境の提供、スマートフォンなどのアプリケーションのクロスプラットフォーム化が挙げられる。また、プロトタイプの作成も継続して行う。

今回ターゲットにした、中高生以外にも全世代で利用できるものとなっている。あまり IT に詳しくない親世代やシニア世代と一緒に使うことも推奨する。ただ、パスワードの複雑化により、攻撃者は別の攻撃手法を用いることもある。近年、身近で増加している 1 つに、フィッシング攻撃(本物を装い、ID やパスワード、その他重要情報を盗み取ろうとする攻撃手法)が挙げられる。その手口も巧妙化している。今後は別の攻撃手法の体験型の教材を作る必要もあるだろう。