HITACHI

Hitachi Systems Security Journal

VOL.79

株式会社 日立システムズ

Hitachi Systems Security Journal VOL.19

TABLE OF CONTENTS

●はじめに
本文書は、株式会社日立システムズ の公開資料です。バックナンバーは以下の Web サイトで確認できます。
https://www.hitachi-systems.com/report/specialist/index.html

教育と実務を滑らかにつなぐ
情報危機管理コンテスト
企業の専門家と創り上げた20年の歩み

川橋 裕 インタビュー

Yutaka Kawahashi

毎年初夏、和歌山県白浜町で開催される学生向け競技大会「情報危機管理コンテスト」。全国の学生がチームを組み、リアルタイムなサイバーインシデントへの対応力を競う本大会は、20 年以上の歴史を持つ。 一般的な CTF とは異なり、最大の特長はシステムの復旧技術だけでなく、関係者への「報連相」や状況判断といったソーシャルな要素を重視する点にある。近年は参加者が生成 AI を駆使する前提の複合的なシナリオが用意され、AI 時代に求められる「人間の判断力」が問われるようになった。また、企業の実務家が運営に協力し、現場のリアルを反映した極めて実践的な環境が提供されている。本稿では長年運営をけん引してきた和歌山大学の川橋裕氏にインタビューし、コンテスト誕生の裏側から独自の評価体制、知識を実践力に変えるセキュリティ人材育成について話を伺った。

取材・文・撮影＝斉藤 健一

情報危機管理コンテストとは

「情報危機管理コンテスト」は、毎年初夏に和歌山県白浜町で開催される「サイバー犯罪に関する白浜シンポジウム」に併設される学生向け競技大会である。実践的なインシデント（システム障害やサイバー攻撃）対応力を養成し、セキュリティ人材を育成することが目的だ。全国の学生が 4 人 1 組のチームを組み、顧客企業のサーバー管理者として、リアルタイムに発生するトラブルへの対応力を総合的に競い合う。

コンテストは段階的に実施される。今年は全国から 15 校 25 チームが参加し、まずは「一次予選」に挑んだ。セキュリティコンサルタントの立場で状況判断やアドバイスを文書化する形式で、近年は生成 AI を活用するチームも増えている。

一次予選を通過したチームは、遠隔から実環境に接続する「二次予選」へと進み、リアルタイムのインシデントレスポンスが求められる。ここを勝ち抜いたチームが、白浜での「決勝戦」へと進出する。

●川橋 裕（かわはし・ゆたか）

工学博士。和歌山大学 学術研究センター 講師。専門分野は、情報セキュリティ、デジタル・フォレンジック、サイバーセキュリティ。受賞歴：2016 年度 情報セキュリティ文化賞（情報セキュリティ大学院大学・日本経済新聞社）、2008 年度 経済産業大臣賞（サイバー犯罪に関する白浜シンポジウム実行委員会）。令和 8 年度（2026 年）文部科学大臣表彰 科学技術賞 理解増進部門

決勝戦の模様をお伝えする前に、コンテストの立ち上げから 20 年以上にわたり運営をけん引し、学生スタッフとともに大会を支え続けてきた和歌山大学学術情報センター・川橋裕氏のインタビューをお届けする。

「攻撃」から「対応力」を競う場へ ―情報危機管理コンテストの誕生と競技の設計思想

斉藤（以下 S）：まず、情報危機管理コンテストがどのような背景で立ち上がったのか教えてください。

川橋（以下 K）：このコンテストはシンポジウムの 10 周年記念企画として始まりました。私は初期から関わっており、立ち上げの経緯を話せる数少ない一人です。もともと和歌山大学では、会場の一角で「裏プログラム」として、攻撃手法や不正挙動を再現するデモを行っていました。

S それは、攻撃の仕組みを見せるような内容だったのですか。

K はい。例えばマルウェアが実行されるとどうなるか、ネットワーク上の挙動や、防御製品がどう反応するかなどを、デモ環境で見せていました。そうした実演は非常に関心を集め、参加者の反応も大きかったですね。

S かなり注目度の高い企画だったわけですね。

K ええ。ただ当時は警察や自治体関係者が多く、攻撃技術を競うコンテストを正面から打ち出すのは難しかったんです。優れたクラッカーを表彰する構図は、日本のこの場にはなじまないと考えました。

S そこで発想を転換したわけですね。

K はい。攻撃を競うのではなく、攻撃を受けた際の防御や危機対応を競う形にひっくり返したんです。それが本コンテストの出発点です。デモで培った知見を別の形で生かす方向に振り直したとも言えます。

S ただ、実際に競技として成立させるのは簡単ではなさそうです。

K そのとおりです。コンテストである以上、シナリオは確実に発火しなければいけません。起きるか分からない不確実なものでは競技にならないため、綿密な検証が必要でした。初期は、どう同じ条件で発火させ、参加者全員にフェアにできる

か、ずいぶん試行錯誤しましたね。

S 運営しながらルールやノウハウも積み上がっていったのですね。

K そうです。回を重ねるごとに、シナリオ設計のノウハウだけでなく、コンプライアンス等のルールも整いました。例えば、どこまでの操作を許容し、何をしてはいけないかという線引きも、運営経験から少しずつ明確になったんです。その積み重ねが、今のコンテストの基盤です。

S 競技自体をどう設計してきたか聞かせてください。インシデントレスポンスを競技にするとなると、一般的な CTF とはかなり発想が違いますよね。

K そうですね。当初から意識していたのは、攻撃の巧みさではなく、事態をどう把握し、判断し、対処するかを見る競技にすることでした。単に原因特定で終わらず、限られた時間で状況を整理し、対応の優先順位を判断し、必要な説明や調整まで行なう。その一連の流れを競技として成立させる必要があったんです。

S 一般的な防御系競技と比べたとき、このコンテストの独自性はどこにありますか。

K 大きな違いは、ソーシャルな要素を重視している点です。単なる技術的な復旧だけでなく、サーバーの再起動や設定変更の可否、誰に説明し許可を取るかというプロセスまで競技に含まれます。実際の対応でも、勝手に手を動かすだけでは済まない場面が多いですよね。だから本コンテストでも、判断、報連相、関係者調整までを含めて対応力として見ています。その意味で、純粋な CTF とはかなり違う競技になっていると思います。

S 短時間の中で、技術だけでなく判断や説明まで求められるわけですね。

K そうです。しかも限られた時間の中で行なうため、現場に近い緊張感があります。技術的な正しさだけでなく、何を優先し、どこまで踏み込み、どう伝えるかを含めて競うからこそ、このコンテストはインシデント対応教育として意味があると思っています。

コンテストを支える独自の運営・評価体制

S ここからは、コンテストを支えてきた運営の体制について聞かせてください。まず、参加校はどのように集めてきたのですか。

K 主に大学や大学院など学術機関の学生が対象です。募集は大きな制度の枠組みというより、シンポジウムを通じた各大学への案内や、つながりのある先生方への声かけなど、人のネットワークに支えられ広がってきました。教育プログラムが体系化される前から続いており、初期は特に教員同士のつながりが大きかったですね。

S 初期は、競技環境そのものを作るのもかなり大変だったのではありませんか。

K ええ、初期は物理的な運営が本当に大変でした。大学内で完結しないため、競技環境をそのまま会場へ持ち込まなければならなかったんです。ミドルタワー PC を何十台も運び、現地で組み上げ、終われば持ち帰るという力仕事でした。その後、外部の教育プロジェクトに関わり設備が少しずつ充実し、ラックやサーバー等も整いましたが、しばらくは大学と会場間で機材ごと移動する運営が続きました。

S 近年は企業や外部の専門家も運営に関わっているそうですが、それはどういう形なのでしょうか。

K 今は実務に近い知見を持つ企業の方々にも協力いただいています。シナリオ作成や運営を共にする中で、学生には外部の実務家と直接議論できる貴重な機会となり、運営側にとっても学ぶことが多いんです。企業側も一方的な支援というより、教育や研修の延長として関わってくださっており、持ちつ持たれつの関係ができていると思います。

S 実際の競技運営では、参加チームごとにかなり細かなサポート体制を敷いているのですよね。

K そうです。決勝ではチームごとに担当スタッフがつき、最低 3 名体制で支えます。攻撃を実行する役、参加者の操作や対応を記録する役、仮想企業の責任者として許可や説明を受ける役など役割が分かれています。参加者の対応を見ながら過程を記録していくため、単に裏方として支えるだけでなく、競技そのものを成立させる重要な体制になっているんです。

S こうした体制を長く続けてこられた背景には、どんな要因があるのでしょうか。

K 特定の大きな組織に過度に頼らなかったことも、継続できた理由の一つです。外部の協力は不可欠ですが、皆さんは前面に出ず、黒子的に支え

てくださっています。特定組織の色が強すぎると、支援が途切れた際に継続が困難になるかもしれません。その点、本コンテストは多くの方に支えられつつ運営の主体性を保ってきました。それが長く続いた理由だと思います。

S インシデントレスポンスの競技は、単純な得点化が難しいと思うのですが、どのように評価しているのでしょうか。

K そこはまさに本コンテストの難しいところです。CTF のように攻撃の成否だけで点数化するわけにはいきません。復旧の速さや正確さはもちろん、参加者がどう状況を整理し、関係者とどうやり取りしたかといった過程も評価対象です。実際に審査する先生方には、そうした過程が伝わる材料をきちんと渡さなければならないため、その材料集めも運営の重要な仕事になっています。

S 技術的な対応力だけでなく、説明の仕方やコミュニケーションもかなり見ているのですね。

K ええ。電話や対面のやり取りで、対応の丁寧さや説明の分かりやすさ、適切な報告タイミングといった要素は、印象を大きく左右します。インシデント対応では技術的に正しくても、説明が不十分だと周囲が動けないことがありますよね。ですから本コンテストでも、参加者と接するスタッフの観察やヒアリング結果を審査に反映させています。単なる復旧の早さだけでなく、対応全体の質を見る考え方です。

S こうした対応の真摯さも評価の軸になっているんですね。

K はい。受賞枠は大臣賞のような大きな賞に加え、協賛各社やその年ごとの賞もあります。そのため 1 チームが複数受賞することもあり、各チームの良さを拾い上げやすい構造になっています。私はこの点を大事にしており、単に順位という一本の物差しで決めるのではなく、チームごとの良さや強みを評価したいんです。そのためにも、審査側に渡す材料はできるだけ多面的でなければならないと考えています。

知識を「実践力」へ変える持続型学習 ―教育と実務の接続をめざして

S 情報危機管理コンテストには、同じ大学や同じ学生が複数年にわたって参加することもあるそうですね。問題の趣向は変えているのでしょうか。

K あります。そのため一度使ったシナリオは 2 ～ 3 年は使いにくくなります。大学内で経験が蓄積し、先輩から後輩へノウハウが伝わるからです。実際、例年参加する大学ほど有利になり、強豪校が出てくるのも自然でしょう。ただ、それを悪いことだとは思っていません。むしろ経験が継承されること自体が教育の一部だと考えています。

S 継承は参加者だけでなく、運営側にもありますよね。

K そのとおりです。運営側も先輩から後輩へ知見を渡さないと続きません。約 20 年続いた背景には、参加校だけでなく運営側にも継承の仕組みがあったからだと思います。コンテストを毎年回すこと自体が、知識や経験を次世代へ受け渡す営みになっているんです。

S 和歌山大学での教育実践としては、どのような考え方を大事にしているのでしょうか。

K 知識を学ぶことと、それを実際の場で使えるようになることは別だと考えています。前者は知識の内化、後者は外化です。大学教育は知識の習得に比重が置かれがちですが、インシデント対応の分野では、どこでどう使うかを経験しないと身につきません。うちのセンターは実際のインフラやトラブル対応の文脈があり、本来は外化の経験を積みやすい環境です。ただ、大学のインフラを学生が自由に触るわけにはいかないため、代わりの実践の場として本コンテストが重要な意味を持っています。

S コンテストが単発のイベントではなく、日常的な教育の延長にあるということですね。

K そうです。コンテストだけで完結せず、SecCap※の授業や社会人向け講座などと連動し、年間を通じて実践の機会が続いています。一般に

※ **SecCap**：enPiT-Security の略称。高度 IT 人材の育成をめざす教育プログラム「enPiT」におけるセキュリティ分野の取り組み。14 の大学が連携し、産業界とも協力して、サイバー脅威の理解とリスクマネジメントに必要な知識、実践力を備えた人材の育成をめざす。

シンポジウムとは別会場で行われた綿情報危機管理コンテスト決勝戦の様子

PBL（Project Based Lerning）と呼ばれる実践型学習がありますが、1回限りの演習ではなく継続的に行なうことが大切です。常に実践の場があることで学生は知識を使い続けられ、経験が先輩から後輩へ蓄積します。その意味で、これを持続的な実践型学習だと考えています。

S インシデント対応教育では、防御だけでなく攻撃側の理解も必要だという考えもお持ちですか。

K はい。直し方や守り方を知るだけでは、インシデントを本当には理解できないと思っています。どんな攻撃があり、どこが狙われ、何をされると弱いかを知らなければ、防御の本質も見えません。ただ、これは扱いの難しい教育でもあります。攻撃の知識は諸刃の剣であり、単に教えればよいわけではありません。だからこそ攻撃だけでなく防御や追跡も同時に学び、全体として理解することが大切だと考えています。

S その意味では、教育の場そのものにも条件が必要になってきますね。

K そう思います。こうした教育は一律に広げるのではなく、目の届く範囲や信頼関係の中で行なうことが重要です。本質的には研究室のような近い関係でこそ可能な教育であり、地方大学など人との距離が近い場の方が規範も働きやすいと感じています。メンターや先輩後輩の関係の中で学ぶことが、スキルを高めるだけでなく、逸脱を防ぐ意味でも大切なのです。

S 最後に、教育と実務の接続について伺います。こうしたコンテストで培われる判断力や調整力は、実際の現場にもつながっていくものなのでしょうか。

K 企業の現場でどれほど役立っているか、卒業後まで細かく追えているわけではありません。ただ、学生の就職先を見ると、セキュリティ関連の仕事に進む例はかなり多いですね。コンサルティングや通信、セキュリティ企業など進路は違いますが、大学時代の実践経験が何らかの形で土台になっていると感じています。

S 卒業生の進路という意味でも、教育の成果は一定程度見えているのですね。

K そう思います。就職先がすべてではありませんが、地方大学の小さな研究室からでも、セキュリティ専門職やコンサル分野に進む学生が毎年出ているのは1つの結果だと思います。単なる知識だけでなく、トラブルに向き合い状況を整理し、人と調整して動いた経験は、現場での大きな強みになるのではないでしょうか。

S 実際の現場で起こるインシデントは、1つの技術知識だけで乗り切れるものではありませんね。

K まさにそこが大事です。対応力と言っても、実際はどんな攻撃でどこを通られ、何をされると弱いのかを複合的に理解しなければなりません。そうした力は座学では身につきにくいため、実戦に近い形で再現し、複数の視点を行き来して考える経験が必要です。企業の実務家とシナリオを考え、現場感のある設定に触れることで、学生の学びはより立体的になると思います。

S 今後の人材育成という意味では、大学の中だけ

シンポジウム 2 日目にはコンテストの総評と表彰式が行われた

で完結しない仕組みも重要になりそうです。

K そうですね。大学生のうちからもっと社会人と触れ合うことが大切です。学ぶ相手が教師だけでは、どうしても世界が狭くなります。イベントや実務に近い形で仕事を経験する機会があってもよい。インターンに限らず、実際の業務の一端に触れて学ぶ仕組みがあれば、学生にとって大きな意味があります。大人と接することで、技術だけでなく働くこと自体を立体的に理解できるからです。

S 教育の延長として、就業体験そのものを位置づけていく発想ですね。

K ええ。そうした学びがもっと増えてよいと思います。全員に同じ形で与えるのは難しいですが、社会に出る前に実務に近い環境に触れる価値は大きいです。特にサイバーセキュリティ分野は机上で見えないことが多いため、教育と実務が滑らかにつながる場を今後もっと作っていく必要があると考えています。

S ありがとうございました。

現場のリアルを再現！
実践的なシナリオで白熱した決勝戦

インタビューに引き続き、白浜で行われた決勝戦の模様をお伝えする。今年は、情報危機管理コンテストとシンポジウムが、それぞれ別会場での開催となり、シンポジウム会場へは中継配信が行われた。

決勝戦には、一次・二次予選を勝ち抜いた 5 チーム（静岡大学、横浜国立大学、名古屋工業大学、大阪工業大学、神戸大学）が駒を進めた。競技では、みずほフィナンシャルグループと AWS が和歌山大学の学生と協力してシナリオを作成し、午前と午後に分かれ現場のリアルな課題を反映した高度な競技が実施された。各シナリオ作成に携わった企業の実務家自らがサポートデスクを担当し、プレイヤーの質問に対応するなど、極めて実践的な環境が用意された。

●午前の部シナリオ：みずほフィナンシャルグループ協力

EC サイトにおけるインシデント対応が課題となった。開発フレームワークの React および Next.js のぜい弱性を突かれ、顧客情報の漏えいが発覚。さらに調査すると、バックドアの設置や漏えい情報を用いた大量の空注文なども判明する複合的な事象であった。プレイヤーは技術的な調査と対応を進めると同時に、経営層への報告書も作成しなくてはならず、技術力とビジネススキルの両方が問われた。

●午後の部シナリオ：AWS 協力

AWS 上の架空決済サービス「SHIRAHAMA Pay」の開始前デモを想定したシナリオが展開された。PQC（耐量子計算機暗号）への対応検討に加え、アクセス集中によるエラーに対処すべく自動スケーリング設定を行なうなど、クラウド特有のトラブルシューティング能力や最新技術への適

応力が試された。

●コンテスト結果

シンポジウム2日目に総評と表彰式が行われた。各チームの技術的なインシデント対応力に加え、コミュニケーション能力や調整力などが多角的に評価され、以下の各賞が授与された。

・経済産業大臣賞：IT ソルーション室 _ げんこつ（静岡大学）
・文部科学大臣賞：HamaGlitch（横浜国立大学）
・JPCERT/CC 賞：P01TERGEIST　奥井 真二郎 氏、滝本 那奈 氏（名古屋工業大学）
・チャレンジ賞：KAGIYA（大阪工業大学）
・ベストコミュニケーション賞：P01TERGEIST（名古屋工業大学）
・インスピレーション賞：やってなんぼ研究（神戸大学）
・みずほフィナンシャルグループ賞：HamaGlitch（横浜国立大学）
・AWS 賞：IT ソルーション室 _ げんこつ（静岡大学）

サイバー脅威がさらに高度化する中、知識を「実践する力」へと変え、社会を支える逞しいセキュリティ人材を輩出し続ける本コンテストの役割は、これからますます重要性を増していくことだろう。

斉藤 健一（さいとう・けんいち）ハッカー専門誌「ハッカージャパン」の創刊から休刊まで（1998 年〜 2013 年）、編集長を務める。現在は、本誌の編集制作を担当する傍ら、サイバーセキュリティ国際会議「CODE BLUE」や、日本ハッカー協会が主催する「Hack Fes.」の運営にも携わっている。

社会のさまざまな動向を把握し、リスクの変化に対応したセキュリティ体制を構築

Hitachi Systems CSI (Cyber Security Intelligence) Watch 2026.06

文＝日立システムズ

ゼロトラストの誤解と意思決定者が検討すべき論点

【概要】ゼロトラストへの関心が高まる中、特定製品の導入で実現できるとする理解が見られる。しかしこれは個別技術や接続の問題と捉えるもので、背景の設計原則を反映していない。本稿ではこの認識のずれを整理し、意思決定者が検討すべき論点を明らかにする。

【本文】ゼロトラストへの関心が高まる一方、実務では「ゼロトラスト対応」製品の導入で対応完了とするケースが見られる。背景には企業側と情報発信側双方の要因がある。企業内にはログインがばらばらなシステム、固定化されたネットワーク、特定担当者への権限集中が残存し、設計原則の全体的な適用が困難な状況にある。一方、「ゼロトラスト」は技術概念であると共に製品販売でも広く使われる。結果、意思決定が「何を導入するか」に偏り、「何を設計すべきか」という本来の論点が後退しやすい。この傾向は規模や業種を問わない。

NIST SP 800-207 が示すとおり、ゼロトラストの本質は位置に基づく暗黙の信頼を排し、利用者・端末・資源・条件で都度判断する設計原則にある。基本要素は最小限の許可、条件別判断、継続的見直し、状況把握である。特定製品で一度に実現できず、複数仕組みを組み合わせ段階的に整備する。この差異を宿泊施設で解説したのが下図である。

なお、技術対策があっても、権限過剰、管理の不透明性、監視の空白が残り、投資がリスク低減に直結しない。問題は技術ではなく、製品導入をゴールとし、継続的運用の設計を欠く点にある。実務では、認証基盤統合、多要素認証、端末状態考慮、資源別アクセス設定、権限見直し、到達範囲制御、ログ統合監視等を状況に応じ並行して進める必要がある。これらは個別対策ではなく上記考え方に沿う。各施策と原則の対応を明確化し、全体整合性を保つことが重要だ。

したがって、意思決定者の論点は製品選択ではなく、誰がどの条件で資源にアクセスするかの方針定義と維持にある。この判断は業務の必要性と許容リスクに基づくため、組織主体の設計が必須だ。ゼロトラストはIT 部門の課題であると共に経営のガバナンス課題でもある。

ゼロトラストセキュリティを宿泊施設をモデルに解説

注：ゼロトラスト解説のため宿泊施設をモデル化したもので、宿泊施設の安全管理を言及したものではありません。

セキュリティツールを実践的に紹介する連載企画

Let's try 不審メール解析

2. メール解析手順編

文＝日立システムズ

1. はじめに

本稿は、各種セキュリティツールなどを実践的に紹介する連載企画です。前号より、「不審メール解析」と題して、受信した不審なメールについて、安全に解析を行うための基本的な手順について整理しています。本書では、不審メールを解析するための環境およびツールの準備を行い、メールを受信したという想定で不審メールに対する一連の解析の流れを整理します。また、攻撃者が使用する不審メールの手口についても整理することで、不審メールの違和感に気づく力などを醸成します。

1. **攻撃手口確認編**
2. **メール解析手順編**

「2. メール解析手順編」では、「1. 解析準備編」で準備を行なった環境を用いて、実際に不審メールの作成を行ない、受信したメールに対して一連の解析を実施することを予定しています。

なお、本稿の安全性には留意していますが、安全を保証するものではありません。OA 端末（社内ネットワーク接続機器）で実施するのではなく、分離された回線内および機器を利用する事を推奨いたします。また、記載されている手順は意図的に無害化・簡略化していることをあらかじめご了承ください。

2. 解析用のメール準備

前号で準備した解析環境を引き続き利用します。新たに構築する場合は、前号の「2. 解析環境の準備」を参考に Windows サンドボックス上に解析環境を準備してください。

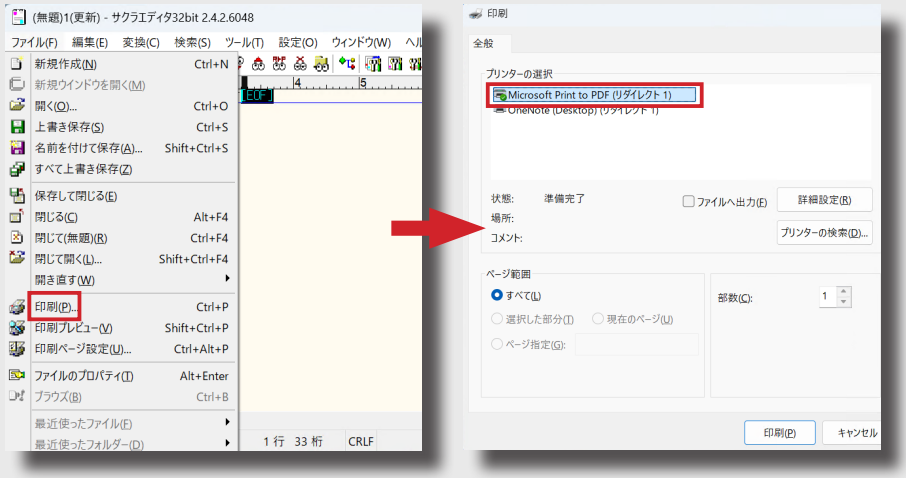
今回の演習では、解析結果を比較するため、不審なメールと正常なメールの 2 つを作成して解析を実施します。なお、本稿では、送信元が偽装されているものや、不審なサイトに誘導することを目的とした仕掛けが施されているメールを、不審なメールの例として取り扱っています。

2.1 正常なメールの作成

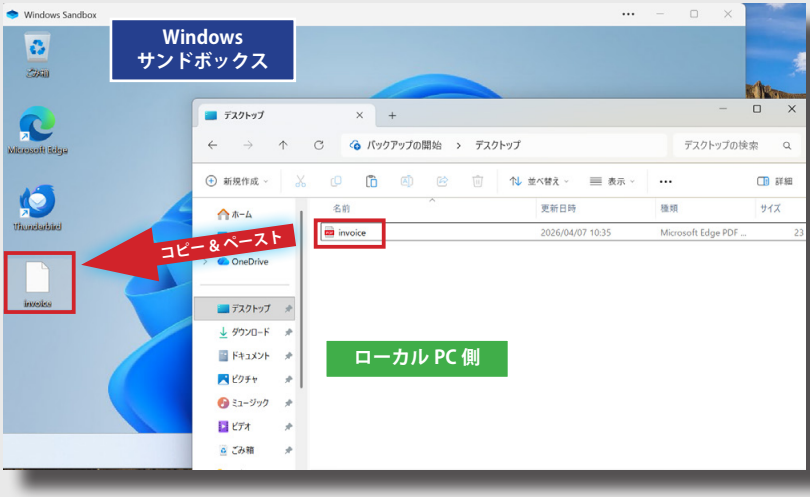
まず、不審メールの解析結果と比較できるように不審なファイルの添付や不審な遷移先を指定していないメールを準備します。 先に添付する PDF ファイルを作成します。サクラエディタを起動し、以下のように記載してください。

```
これは解析用のダミーファイルです
```

入力を終えたら図のように「ファイル」-「印刷」を選択し、プリンターが「Microsoft Print to PDF」を選択していることを確認してから印刷ボタンを押して、「invoice.pdf」という名前で PDF ファイルを保存します。

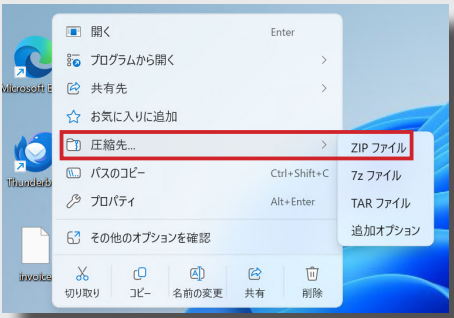


設定によっては、印刷先を Windows サンドボックス上に指定できないことがあります。この場合は、Windows サンドボックス外のローカル PC 側を指定して保存します。その後、当該ファイルを Windows サンドボックスのデスクトップにコピーしてください。

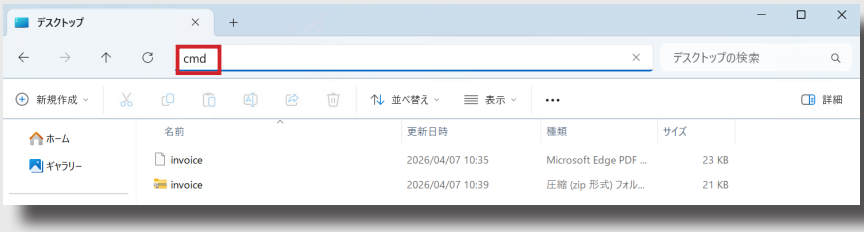


侵害の原因となる悪意あるメールでは、マルウェアなどを圧縮した ZIP ファイルを添付し、受信者に開かせる手法がよく用いられます。次項では、こうしたケースを想定した不審メールを作成しますが、本ハンズオンでは比較調査を行なう目的で、同様の手法を用いた正常なメールも作成します。

そのため、まずは「invoice.pdf」ファイルを右クリックし、「圧縮先」から「ZIP ファイル」を選択して、デスクトップ上に「invoice.zip」として保存してください。



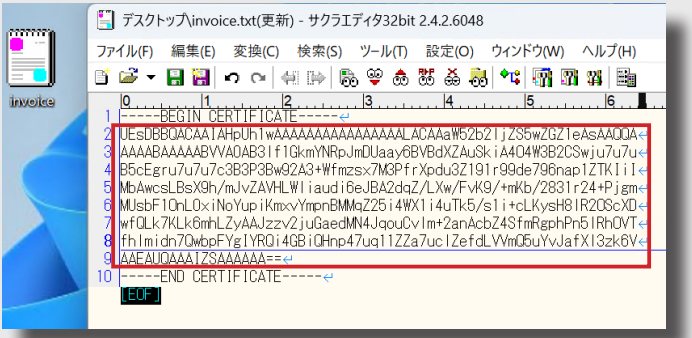
次に、メールに添付する準備として、ファイルを Base64 に変換します。エクスプローラーで「invoice.zip」の保存フォルダーを開き、アドレスバーに「cmd」と入力してコマンドプロンプトを起動してください。



コマンドプロンプトが起動したら、以下のコマンドを実行してください。

```
certutil –f –encode “invoice.zip” “invoice.txt”
```

コマンドを実行すると、「invoice.zip」と同じデスクトップ上に「invoice.txt」が出力されます。このファイルを開くと、下の図のように「BEGIN CERTIFICATE」と「END CERTIFICATE」で挟まれた文字列が確認できます（可読性確保のため、図では文字列を省略しています）。



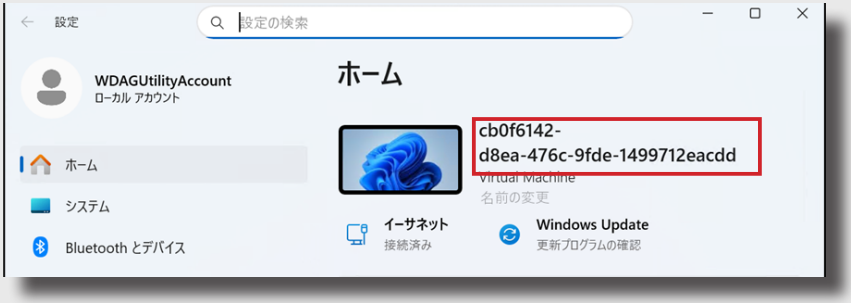
前ページの図で赤枠で囲まれた「BEGIN CERTIFICATE」と「END CERTIFICATE」の間の部分は、後ほどメール送信時に使用するため、コピー＆ペーストできるようテキストを開いたままにしてください。この文字列は、後述の「message.txt」内の指定箇所に貼り付けます。
次にサクラエディタを起動して以下の内容をコピー＆ペーストし、デスクトップ上に「message.txt」として保存してください。なお、デファング（[:]、[.]）されている箇所は、[ ] を外して記載してください。
また、「Received:」行の改行後には TAB または半角スペースが必要ですが、コピー＆ペースト時にこれらが欠落する場合があります。その際は、ご自身で修正してください。

```
Received: from relay2.example[.]net (203.0.113[.]2)
      by “Windows コンピューター名 ” (“Windows コンピューター名 ” [::1]) with SMTP
      for <strike@hitachi-systems[.]com>; Wed, 1 Apr 2026 10:10:00 +0900 (JST)
Received: from relay1.example[.]net (203.0.113[.]1)
      by relay2.example[.]net with SMTP
      for <strike@hitachi-systems[.]com>; Wed, 1 Apr 2026 10:05:00 +0900 (JST)
Received: from client.example.co[.]jp (203.0.113[.]10)
      by relay1.example[.]net with SMTP
      for <strike@hitachi-systems[.]com>; Wed, 1 Apr 2026 10:00:00 +0900 (JST)
From: billing@example.co[.]jp
To: strike@hitachi-systems[.]com
Subject: Check your billing details
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="BOUNDARY"
（コピー＆ペースト後、この文字列を削除して空行にしてください）
--BOUNDARY
Content-Type: text/html; charset=UTF-8
（コピー＆ペースト後、この文字列を削除して空行にしてください）
<html>
 <body>
  <p> サービスのご利用ありがとうございます。</p>
   <p> ご購入いただいた製品に関する請求書を本メールに添付しておりますのでご確認ください。</p>
   <p> もし心当たりがない場合は、以下のボタンからマイページにログインいただき、購入履歴をご確認ください。</p>
    <a href="https[:]//login[.]example[.]co[.]jp" style="background-color: #ff9900; color: white; padding: 10px 20px; text-decoration: none; border-radius: 5px;"> マイページにログインする
</a>
 </body>
</html>
（コピー＆ペースト後、この文字列を削除して空行にしてください）
--BOUNDARY
```

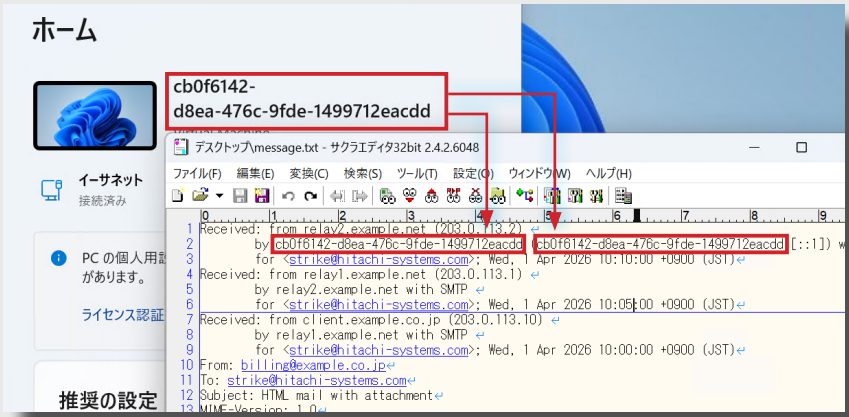
次ページに続く

Content-Type: text/plain; name="invoice.zip"
Content-Disposition: attachment; filename="invoice.zip"
Content-Transfer-Encoding: base64
（コピー＆ペースト後、この文字列を削除して空行にしてください）
（ここに「invoice.txt」内の「BEGIN CERTIFICATE」と「END CERTIFICATE」で挟まれた Base64 文字列を貼り付け）
（コピー＆ペースト後、この文字列を削除して空行にしてください）
--BOUNDARY--

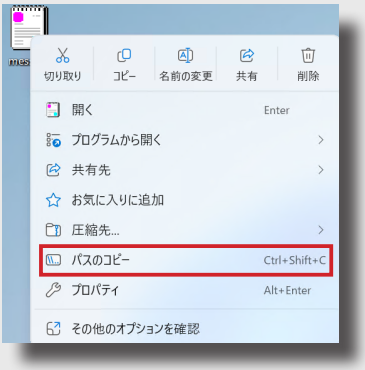
作成したテキスト内の「Windows コンピューター名」には、スタートボタンから「設定」を開き、ホーム上部に表示されるコンピューター名を入力してください。
また、「設定」の「システム」内にある「バージョン情報」から、デバイス名をコピーして使用することも可能です。
なお、筆者の環境では、図の赤枠で示すように「cb0f6142-d8ea-476c-9fde-1499712eacdd」でした。



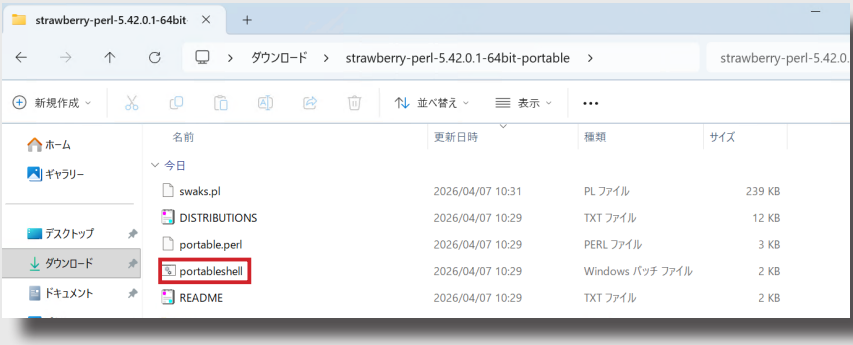
確認したコンピューター名を、「message.txt」内の「Windows コンピューター名」（2 か所）に入力してください。



次に、SMTP サーバーテスト用プログラム「Swaks」を用いて、用意した正常なメールを Mailpit サーバーへ送信します。
まず送信準備として、下図のようにデスクトップ上の「message.txt」を右クリックし、「パスのコピー」から詳細なパスを取得してください。



次に、前号の「2.5 Swaks の準備」で用意した portableshell.bat を実行してください。



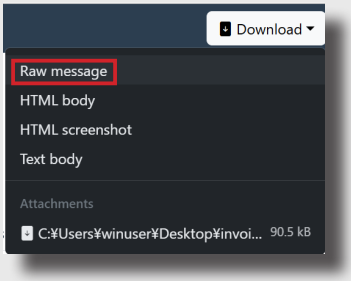
bat ファイルを実行して起動したコマンドプロンプトで、以下のコマンドを実行してください。
なお、「message.txt のパス」の箇所には、先ほど「パスのコピー」で取得したパスを挿入してください。

```
perl ./swaks.pl ^
--from “billing@example.co.jp” ^
--to “strike@hitachi-systems.com” ^
--server “localhost” ^
--port 1025 ^
--data @”(message.txt のパス )”
```

なお、筆者の環境では、パスが「C:¥Users¥WDAGUtilityAccount¥Desktop¥message.txt」だったため、data の行は下図のようになります。

```
--data @”C:¥Users¥WDAGUtilityAccount¥Desktop¥message.txt”
```

コマンドを実行したら、Microsoft Edge のアドレスバーに「localhost:8025」と入力し、Mailpit の WebUI を開いてください。
受信したメールを確認し、「Download」ボタンから「Raw message」を選択して .eml ファイルをダウンロードしてください。

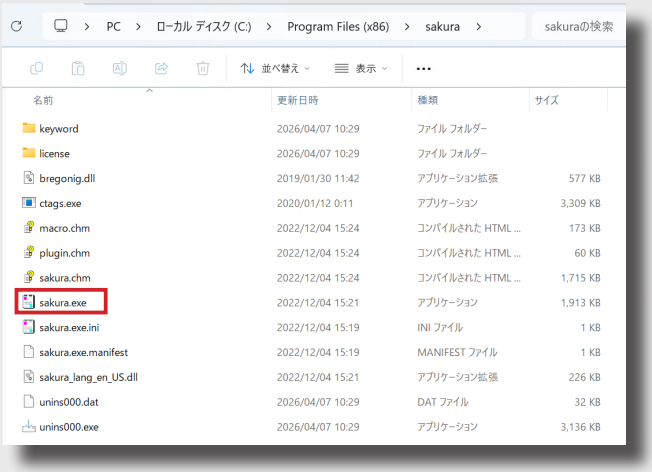


2.2 不審なメールの作成

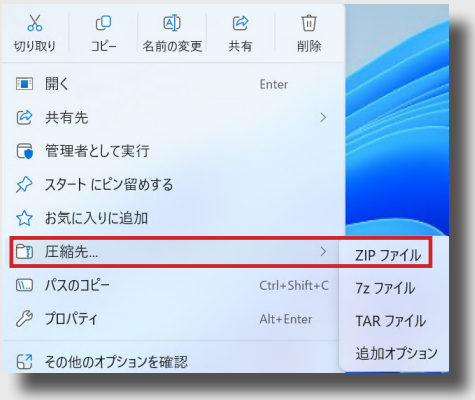
準備した Windows サンドボックス上で、解析対象となる不審メールを作成します。今回はマルウェアが添付されたメールを想定し、無害な実行ファイルを添付ファイルとして代用します。
前号で紹介したとおり、不審メールの添付ファイルは受信者にクリックさせるための偽装が施されていることが多いため、今回は実行プログラムを PDF 形式の請求書に見せかける偽装を行います。
まず、エクスプローラーを起動し、上部のアドレスバーに以下のパスを入力して Enter を押下してください。

```
C:¥Program Files (x86)¥sakura
```

開いたフォルダー内にある「sakura(.exe)」が、サクラエディタの実行ファイルです。似た名前が多く見分けづらい場合は、エクスプローラー上部の「表示タブ > 表示 > ファイル名拡張子」に一時的にチェックを入れると探しやすくなります。



次に、該当のファイルをデスクトップにコピーし、右クリックの「名前の変更」からファイル名を「invoice.pdf.exe」に変更してください。
なお、先ほどファイル名拡張子を表示する設定にした場合は、エクスプローラーの「表示タブ > 表示 > ファイル名拡張子」のチェックを外して非表示に戻してください。
exe 形式のファイルをそのままメールに添付すると、セキュリティ機能に検知されたり、メールソフト独自のアイコンが表示されて偽装が見破られたりする可能性があります。攻撃者はこれを回避するため、exe ファイルを ZIP 形式に圧縮して添付することがあります。本演習でもその状況を想定し、「invoice.pdf.exe」を右クリックして「圧縮先 > ZIP ファイル」を選択し、ZIP 形式で圧縮しておきます。



次に、前節「2.1 正常なメールの作成」の手順を参考にデスクトップ上でコマンドプロンプトを起動し、以下のコマンドを実行して Base64 に変換します。なお、今後の作業を進めやすくするため、ファイルの保存先にはデスクトップを指定してください。

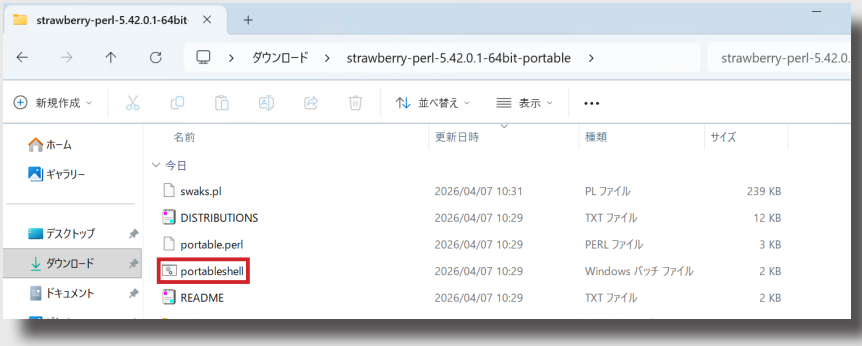
```
certutil –f –encode “invoice.pdf.zip” “invoice.pdf.txt”
```

デスクトップに出力された「invoice.pdf.txt」を開き、前項と同様に「BEGIN CERTIFICATE」と「END CERTIFICATE」で挟まれた文字列を後で貼り付けられるよう、ファイルは開いたままにしておいてください。

次に攻撃者が利用することの多い手口を模倣しながら、先ほど作成した不審ファイルを添付した解析用の不審なダミーメールの作成を行ないます。サクラエディタを起動し、以下の内容をコピー＆ペーストし、「名前を付けて保存」からデスクトップ上に「evil_message.txt」という名前で保存してください。なお、デファング ([:]、[.]) している箇所は [ ] を外して記載してください。今回不審な遷移先として Hitachi Systems Security Journal を指定していますが、実際の不審メールでは該当の URL が記載された場所に不審な URL が記載されることになります。

```
Received: from bad-actor.example[.]net (203.0.113[.]2)
      by “Windows コンピューター名 ” (“Windows コンピューター名 ” [::1]) with SMTP
      for <strike@hitachi-systems[.]com>; Wed, 1 Apr 2026 10:05:00 +0900 (JST)
Received: from relay1.example[.]net (203.0.113[.]1)
      by relay2.example[.]net with SMTP
      for <strike@hitachi-systems[.]com>; Wed, 1 Apr 2026 10:10:00 +0900 (JST)
Received: from client.example.co[.]jp (203.0.113[.]10)
      by relay1.example[.]net with SMTP
      for <strike@hitachi-systems[.]com>; Wed, 1 Apr 2026 10:00:00 +0900 (JST)
From: billing@example.co[.]jp
To: strike@hitachi-systems[.]com
Subject: Check your billing details
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="BOUNDARY"
（コピー＆ペースト後、この文字列を削除して空行にしてください）
--BOUNDARY
Content-Type: text/html; charset=UTF-8
（コピー＆ペースト後、この文字列を削除して空行にしてください）
<html>
 <body>
  <p> サービスのご利用ありがとうございます。</p>
   <p> ご購入いただいた製品に関する請求書を本メールに添付しておりますのでご確認ください。</p>
   <p> もし心当たりがない場合は、以下のボタンからマイページにログインいただき、購入履歴をご確認ください。</p>
    <a href="https[:]//www[.]hitachi-systems[.]com/report/specialist/hj/index[.]html" style="background-color: #ff9900; color: white; padding: 10px 20px; text-decoration: none; border-radius: 5px;"> マイページにログインする </a>
 </body>
</html>
（コピー＆ペースト後、この文字列を削除して空行にしてください）
--BOUNDARY
Content-Type: text/plain; name="invoice.pdf.zip"
Content-Disposition: attachment; filename="invoice.pdf.zip"
Content-Transfer-Encoding: base64
（コピー＆ペースト後、この文字列を削除して空行にしてください）
（ここに「invoice.txt」内の「BEGIN CERTIFICATE」と「END CERTIFICATE」で挟まれた Base64 文字列を貼り付け）
（コピー＆ペースト後、この文字列を削除して空行にしてください）
--BOUNDARY--
```

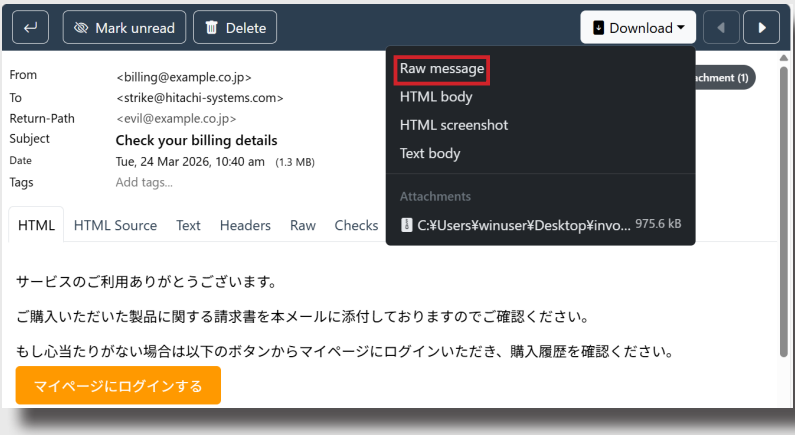
次に、前節のメール送信手順と同様に、Swaks の準備で使用した「portableshell.bat」を実行してください。



bat ファイルを実行して起動したコマンドプロンプトで、以下のコマンドを実行してください。なお、パスの箇所には前節の手順と同様に、デスクトップ上の「evil_message.txt」を右クリックして「パスのコピー」から取得したパスを挿入してください。

```
perl ./swaks.pl ^
--from “billing@example.co.jp” ^
--to “strike@hitachi-systems.com” ^
--server “localhost” ^
--port 1025 ^
--data @”(evil_message.txt のパス )”
```

送信完了後、Mailpit の WebUI で受信したメールを確認し、画面右上の「Download」ボタンから「Raw message」を選択して .eml ファイルをダウンロードしてください。

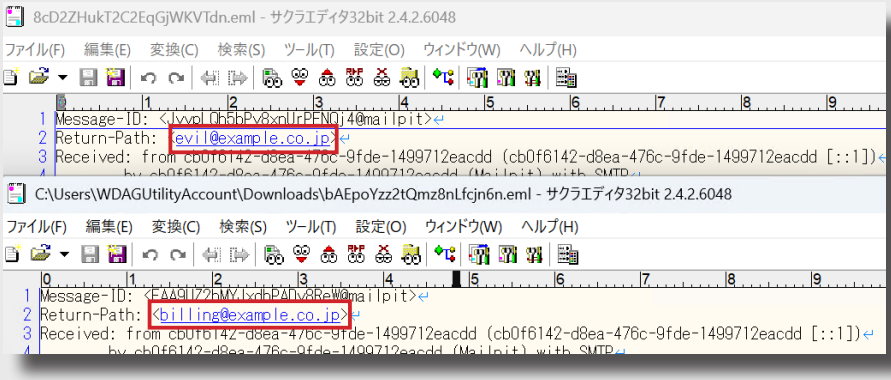


3. メール解析の手順

ここからは、前章で作成した不審メールと通常のメールについて、要点を絞って順番に解析します。

3.1 ヘッダーの解析

まず、ヘッダーの調査を行います。メールヘッダーには送信元や配送経路などの情報が含まれており、これを確認することで不審メールかどうかの判定が可能です。
2 章で準備した 2 通の .eml ファイルをサクラエディタで開き、「Return-Path」を確認してください。不審メールでは、「From」とは異なるアドレスが記載されていることが分かります。



Return-Path は配信エラー時の戻り先アドレスであり、不審メールはこれが From アドレスと異なるため、送信元の偽装が疑われます。
次に、メール配送に関わったサーバーの記録が残る「Received ヘッダー」を確認します。通常、このヘッダーは以下の要素で構成されています。

```
Received:   from [ 送信元ホスト名 ] ([ 送信元 IP アドレス ])
            by [ 受信者側サーバーホスト名 ] ([ 受信者側 IP アドレス ])
            with [ メールプロトコル ]
            id [ 管理 ID]
            for <[ 受取人アドレス ]>；
            [ タイムスタンプ ]
```

経路上の各サーバーはメール受信時に情報をヘッダー上段に追加するため、最下段が発信元、最上段が受信者（自組織）となります。
この情報は改ざんが可能なため、不自然な点がないかの確認が、不審メールを見分ける判断材料となります。以下に確認すべき観点を整理します。

・経路の断絶

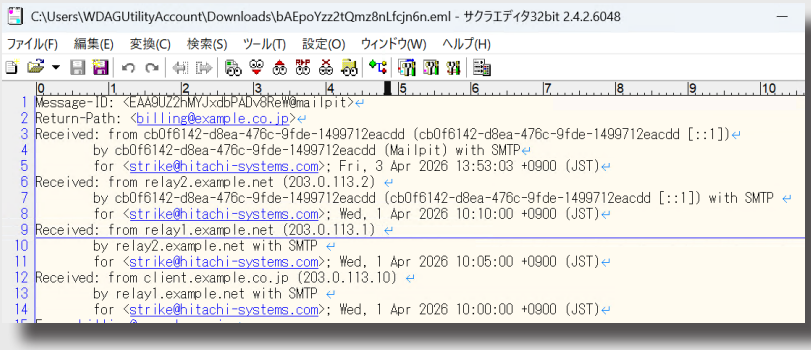
Received ヘッダーの情報が以下のようになっている場合を考えます。

1 段目：from example-C[.]com by example-D[.]com…
2 段目：from example-A[.]com by example-B[.]com…

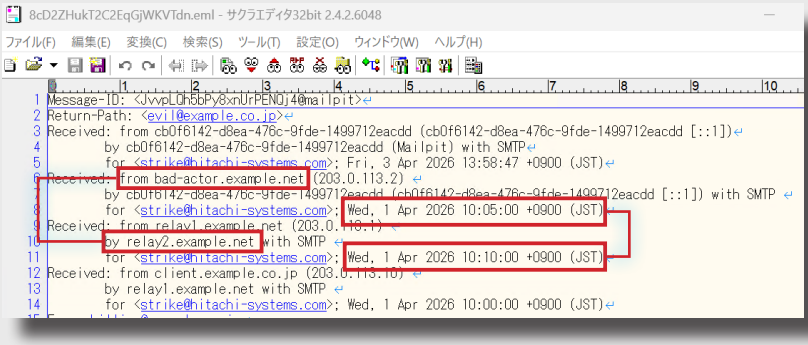
この例では、1 段目は example-C から example-D へ送信されていますが、2 段目では example-A から example-B が受信したことになっています。このように経路がつながっていない場合、攻撃者によるヘッダー情報の改ざんが疑われます。
また、Received ヘッダーに記録された情報をもとに、Whois での所有者確認や、ネット上で悪用情報がないか調査することも有効です。

・時刻の逆転

サーバーを経由した情報はヘッダーの上部に追加されていきます。そのため、上下のタイムスタンプを比較し、下段より上段の時刻が古い場合は、攻撃者による不正な改ざんが疑われます。
これらの情報を踏まえ、まずは作成した通常のメールから確認します。



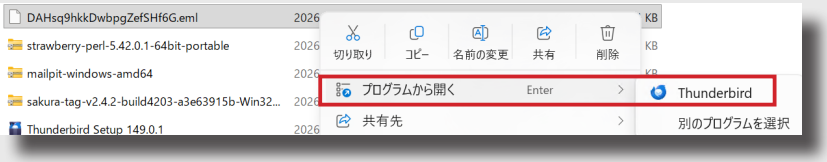
ヘッダー情報を確認すると、client → relay1 → relay2 → Mailpit の順に送信されており、経路の断絶やタイムスタンプの逆転が起きていないことが分かります。
次に、不審メールのヘッダーを確認します。



経路を確認すると、client → relay1 → relay2、bad-actor → Mailpit となっており、2 段目と 3 段目の Received ヘッダー（from と by）に矛盾が生じていることが分かります。
また、Received ヘッダーは下段から上段へ追加されるため、通常は下段の方が古い時刻になります。しかし今回の不審メールでは、2 段目から 3 段目にかけて「10:10 → 10:05」と時刻が逆転しており、攻撃者による情報の改ざんがうかがえます。
ただし、メールサーバーの NTP（時刻同期）設定不備やタイムゾーンの確認漏れなどにより、正常なメールでも時刻が不自然に見える場合があります。そのため、時刻情報だけで不審メールと断定せず、経由した全サーバーのドメイン確認など、他の情報もあわせて総合的に判断することが重要です。

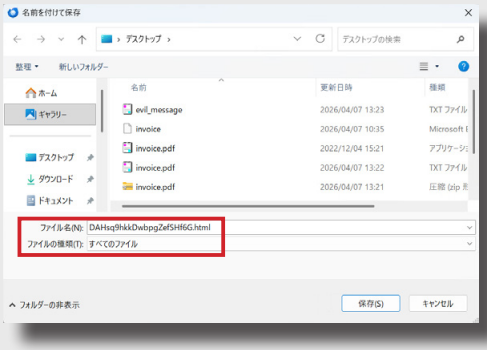
3.2 メール本文の解析

HTML 形式のメールでは、表示上の URL と実際の遷移先が異なったり、偽のログインボタンから不審なサイトへ誘導されたりする場合があります。本文の内容を確認するため、.eml ファイルを Thunderbird で開いてください。

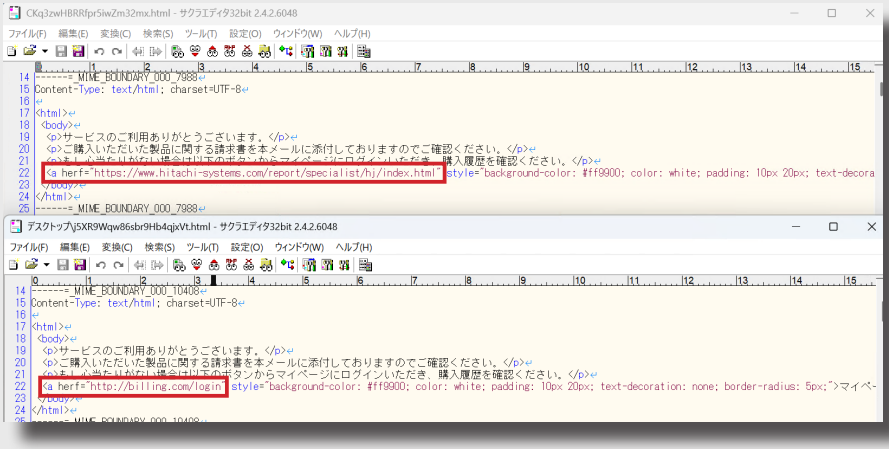


Thunderbird の「ファイル」メニューから、「名前を付けて保存」>「ファイル」を選択してください。

次に、ファイルの種類を「すべてのファイル」に指定し、ファイル名の末尾に「.html」を付けてデスクトップに保存してください。



保存した html ファイルをサクラエディタで開き、本文に記載された URL やドメインを Whois や Web 検索、オンラインサンドボックスなどで検索して調査します（今回は当社のページを指定していますが、実際の不審メールではフィッシングサイトなどへの誘導に悪用されます）。
2 通のメールを比較するとわかるように、サクラエディタで html ファイルのソースを確認することで、不審な遷移先が設定されていないかのチェックが可能です。



また、メールの日本語に違和感がないか確認したり、特長的な件名や本文を Web 検索したりすることでも、不審メールの判断が可能です。

3.3 添付ファイルの解析

添付ファイルを Windows サンドボックス上の任意のフォルダに保存し、ZIP を展開してください。
展開した不審ファイルのプロパティを確認すると、表示名は「invoice.pdf」ですが、実際のファイルの種類は「アプリケーション (.exe)」であることが分かります。

また、以下のコマンドでハッシュ値を調べ、オンラインサンドボックスなどで検索して不審なファイルかどうかを確認することも可能です。

```
certutil –hashfile “ ファイルのパス ” SHA256
```

4. おわりに

今回はここまでです。

「不審メール解析 2. メール解析手順編」では、検証環境で正常・不審なメールを作成してローカルの SMTP サーバーへ送信し、実際の受信時を想定したヘッダー・本文・添付ファイルの解析を行いました。本稿が、不審メールを受信した際の解析手順を理解する一助となれば幸いです。

Hitachi Systems Security Journal

株式会社 日立システムズ　本社：〒141-8672　東京都品川区大崎 1-2-1　www.hitachi-systems.com

お問い合わせは

Printed in Japan