安全解析手法 STAMP/STPA に対するセキュリティ視点からの脅威分析の拡張提案

金子朋子[†1] 髙橋雄志[†3] 大久保隆夫[†2] 勅使河原可海[†3] 佐々木良一[†3]

概要： STAMP （System Theoretic Accident Model and Processes）とその安全分析手法である STPA （System Theoretic Process Analysis）は IoT 時代の複雑なシステムに対する新たな安全性解析手法として注目を集めている．
STAMP/STPA はセーフティを中心に展開されてきたが，セキュリティ上のリスク分析にも適用可能であり，STPA のセキュリティ対応手法である STPA-Sec も提案されている．しかしながら，現在の STPA-Sec はミッション・ビジネスレベルに焦点をおき，セキュリティ本来の視点からの脅威分析には言及していない．そこで STAMP/STPA-Sec の手順の中に脅威分析を追加する提案を行う．具体的には STRIDE モデルを利用したヒントワードの設定と Threat Tree 分類を組み込むことで，システマチックなセキュリティシナリオ作成を可能とする．

キーワード：STAMP/STPA，脅威分析，STRIDE，Threat Tree，セーフティ，セキュリティ・バイ・デザイン

Proposing enhancement of a threat analysis from the security perspective for STAMP/STPA as a safety analysis

KANEKO TOMOKO[†1] TAKAHASHI YUJI[†3] TAKAO OKUBO[†2]
TESHIGAWARA YOSHIMI[†3] RYOICHI SASAKI [†3]

Abstract:
STAMP (System Theoretic Accident Model and Processes) and its safety analysis application, STPA (System Theoretic Process Analysis) have attracted much attention as a new safety analysis method for complex systems of IoT. Though, STAMP/STPA is disseminated as a safety analysis technique, they also can be applied in the security risk analysis, and the security response of STPA is proposed as a STPA-Sec. However, current STPA-Sec is focused on mission and business level, not referring the threat from the security perspective. We propose adding only the threat analysis in the procedure of STPA-Sec. Specifically, using the STRIDE model as “hintwords” and by incorporating the Threat Tree classification, a systematic security scenario is created.

Keywords: STAMP/STPA， Threat Analysis, STRIDE, Threat Tree, Safety, Security by Design

1. †はじめに

現代のシステムはネットワークを介して様々な機器やクラウドと連携しながら動作する新たなサービスが拡大している．このように異なる分野の製品や産業機械などがつながって新しいサービスを創造するモノのインターネット（IoT: Internet of Things）は新産業革命とまで言われ，大きな期待を集めている．IoT は家電，自動車，各種インフラ業者など新規プレーヤーの登場を産み，その取り込みは加速化している．しかし相互につながる際に最も懸念されるのは，IoT システムへのセキュリティ上の脅威である．IoT システムにおいても攻撃者はシステムの脆弱性を突いて攻撃

1 情報処理推進機構 INFORMATION-TECHNOLOGY PROMOTION AGENCY, JAPAN（IPA)

†2 情報セキュリティ大学院大学 INSTITUTE of INFORMATION SECURITY

†3 東京電機大学 TOKYO DENKI UNIVERSITY

を仕掛けてくるためである．

IoT 時代には相互につながるシステムへの脅威に対して，より安全な機器，システムを開発することが必要とされる[1]．そのためには，従来の情報セキュリティ上の機密性と完全性と可用性に加え，安全の視点が必要になる．そこで筆者らは安全の視点でリスク分析を行うために IoT 時代の複雑なシステムに対する新たな安全解析手法 STAMP（System Theoretic Accident Model and Processes）[2][3]とそのハザード分析手法である STPA （System Theoretic Process Analysis） [4][5]に着目し，安全を考慮したセキュリティ分析手法の創出を目指した．

STAMP/STPA は安全（セーフティ）を中心に展開されてきたが，セキュリティを考慮した分析にも適用可能であり[5], STPA のセキュリティ対応手法である STPA-Sec も提案されている[6][7]．しかしながら，現在の STPA-Sec はミッション・ビジネスレベルに焦点をおき，脅威分析には言及していない．そこで本論文では STAMP/STPA-Sec の手順の中にセキュリティ本来の視点からの脅威分析を加える提案を行う．具体的には STRIDE モデルを利用したヒントワードの設定と Threat Tree 分類を組み込むことで，システマティクなセキュリティシナリオ作成を可能とする．

本論文は 2 章で STAMP と各種ハザード手法，セキュリティ要求分析に関連した技術と考え方を紹介する．続く 3 章では現在の STPA のセキュリティ対応について示す．4 章は STPA-Sec の課題を踏まえたうえで脅威分析の追加方式の提案を示す．5 章では STPA への脅威分析追加方式を適用した事例を述べる．5 章で提案方式に関する考察を行い，6 章でまとめと今後の方針について述べる．

2. 関連研究

2.1 STAMP と関連手法

STAMP とはシステム理論に基づく事故モデルであり，STPA は STAMP モデルに基づく代表的な手法であり，ハザード分析を行う．

前提として「システム事故の多くは，構成要素の故障ではなく，システムの中　で安全のための制御を行う要素（制御要素と被制御要素）の相互　作用が働かない事によって起きるとし，「要素（コンポーネント）」と「相互作用（コントロールアクション）」に着目してメカニズムを説明「アクションが働かない原因」が「コントロールアクションの不適切な作用」に等しいという視点を持つことで原因を有限化している．

STAMP に基づく分析の道具立てとプロセスとして，プロセスは仕様記述，安全性ガイド設計，設計原理などのシステム工学，リスク管理の運用，管理の原則/組織設計の規制を利用する（図 1）．

ツール（手法）には STAMP モデルに基づき，事故/イベント分析（CAST: Causal Analysis based on STAMP），ハザード分析（STPA），早期概念分析（STECA: Systems-Theoretic Early Concept Analysis），組織的/文化的リスク分析，先行指標識別，セキュリティ分析（STPA-Sec）が提示されている．事故/イベント分析（CAST）は事故が起きてからイベントとして分析する手法，STPA-SEC はそのセキュリティ版である（図 1）．

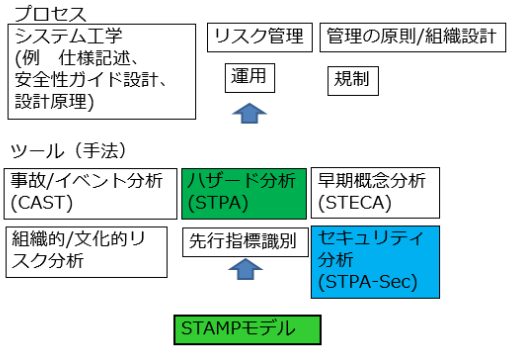

図 1.　STAMP に基づく分析の道具立てとプロセス

セーフティとセキュリティを統合する手法としては STPA-SafeSec が提案されている[8]．

2.2 ハザード分析手法

FTA（Fault Tree Analysis），FMEA（Failure Mode and Effect Analysis）は，フォールトツリー図や影響分析表を用いてハザード要因を分析する伝統的なハザード分析手法である．システムの構成要素と故障モードが決まるアーキテクチャ設計の段階から適用できる．機器や組織の単一故障をハザード要因として識別する分岐条件を論理的に組むことで網羅的に分析できる特徴をもつが，深く分析できる反面，全体的な視野での分析が難しい（図 2）．

STPA は STMAP モデルに基づき，コントロールストラクチャーとコントロールループ図を用いてハザード要因を分析する安全解析手法である．システムの大まかな構成要素が決まる概念設計の段階から適用できる．複数の機器や組織（人間）が，相互作用を行う複雑なシステムにおいて，相互作用に潜むハザード要因を識別する特徴をもち，過去のアクシデント事例データに基づくガイドワードにより網羅的に分析できる．またシステム全体の振る舞いを確認しながら分析できる．

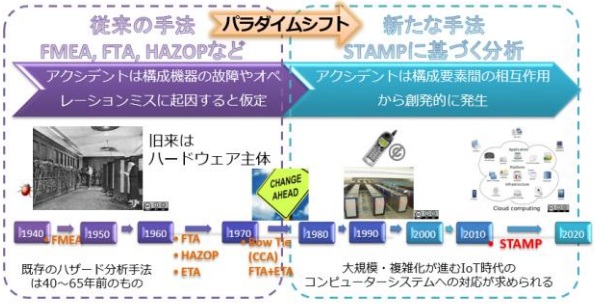

図 2.　STAMP 出現の背景

2.3 「セキュリティ・バイ・デザイン

内閣府サイバーセキュリティセンター （NISC）によると「セキュリティ・バイ・デザイン」とは「情報セキュリティを企画・設計段階から確保するための方策」[9]であり，「安全な IoT システムのためのセキュリティに関する一般的枠組」[1]においては，目的としてまた基本原則として掲げられている重要な概念である．IoT 時代を迎え，セキュリティ上の脅威が多大な被害を及ぼす可能性がでてきているため，企画・要件定義工程や設計工程というより早い段階から事前にセキュリティを作りこむことが求められているのである（図 3）．

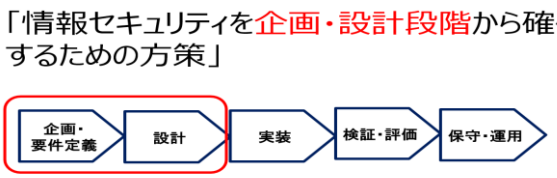

図 3.　セキュリティ・バイ・デザインの定義

2.4 セキュリティ要求分析手法

セキュリティ要求分析では，顧客は要求に基づく機能要件の分析に加えて攻撃者の存在を考慮した非機能要件の分析を必要とする．そこでセキュリティ要求はアセットに対する脅威とその対策の記述が必須となる.セキュリティ要求分析の手法にアタックツリー[10][11]，ミスユースケース[12]，Secure Tropos[13]，i*-Liu 法[14]，Abuse Frames[15]やアクタ関係表に基づくセキュリティ要求分析手法（SARM）[16][17]などがある．いずれの手法もセキュリティを考慮した脅威分析やそれに対する対策立案の手法だが，明示されない非機能要求に関して要件をつくすことは難しいのが実状である．

また SQUARE[18][19]はセキュリティのシステム品質を高めるために定められた特定の手法によらないプロセスモデルである．SQUARE は生産物の定義に基づいてリスク分析し，セキュリティ要求を抽出・優先順位付け・レビューする手順である．

マイクロソフトのセキュリティ開発ライフサイクル[20]はデータフロー図を詳細化し脅威の観点 STRIDE で脅威分析を実施する．設計による安全性確保を重視し設計段階でセキュリティ要求を抽出している．また STRIDE を元にした Threat Tree 分類[21]も示されている．しかしながら IoT セキュリティ要求に最適化した手法はまだ定められてはいない．

3.　STPA のセキュリティ対応

STAMP/STPA は安全性解析手法であり，セーフティを中心に展開されてきたが，これらの特徴はセキュリティ上のリスク分析にも適用可能である．そこで 2016 年 3 月 STAMP workshop でのチュートリアル資料[6]に基づいて，サイバーセキュリティなどに特化した事故分析手法である STPA-Sec について説明する．

①既存のアプローチと比較した特徴

STPA は全体を俯瞰してトップダウンに分析をする手法である．STPA-Sec も同様に全体俯瞰の上で，トップダウンに分析を実施する．STPA-Sec は従来のセキュリティエンジニアリングがフォーカスしてきた物理的な機能（図 4 の青い円の部分）に比べ，より広範囲で概念的な機能・目的にフォーカス（図 4 の緑の縁の部分）している．

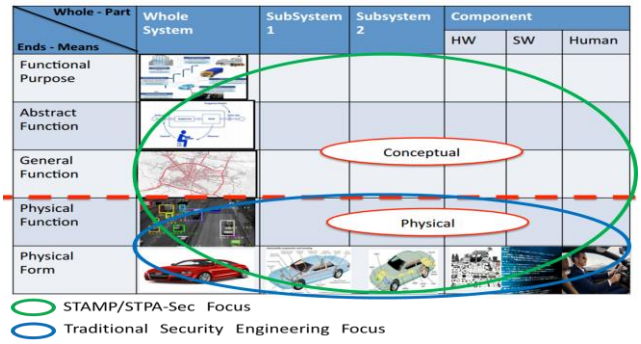

図 4.　STPA-Sec のフォーカスエリア

また，STPA は手段（How）に着目した手法ではなく What に着目した手法である．STPA-Sec も同様に従来のセキュリティ要求分析手法であるアタックツリーやミスユースケースのようにどのような脅威があるのかを洗い出す手段（How）ではなく，コンセプト段階での問題が何（What）であるかを明確にするアプローチである．

セキュリティ対策は現状，保守・運用段階での脆弱性対処が中心である．しかし，多様な機器・システムが複雑につながる IoT 時代には何がセキュリティを確保する際の問題となるのかを事前に対処できることがより重要になってきている．そのため STPA-Sec のアプローチは現在のセキュリティ開発の課題解決に役立つであろう．

②STPA と STPA-Sec との違い

STPA と STPA-Sec の分析手順は，セキュリティ上の脅威抽出に必要な分析の視点が追加されることを除けば，基本的には変わらない．要因の特定に関して図 5 に示すオレンジの部分が STPA-Sec での追加事項となる．コントローラーなどに悪意ある，権限を持たない，部分的なインプットを要因として追加的に分析される．

③STPA-Sec 分析手順

STPA-Sec の分析手順は，非安全と対になるように，セキュアでない（Unsecure）を追加している．つまり，安全でない状態を考えるのと同様に，セキュアでない状態を考慮することになる．また，非安全なコントロールの原因の特定に際し，セキュアでないコントロールアクションを導くシナリオを識別し，影響度を踏まえ，よりクリティカルなコントロール戦略を選択することなどが STPA に対する主な追加事項である．

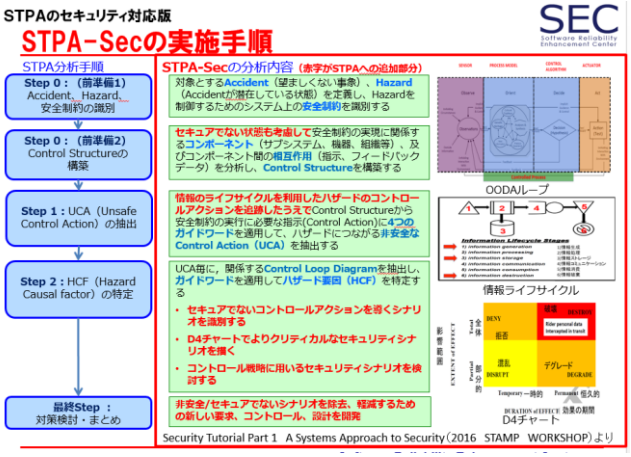

図 5. STPA-Sec の実施手順

4. 脅威分析の追加方式

4.1 現在の STPA-Sec の課題

現在の STPA-Sec はコンセプチュアルなミッション・ビジネスレベルに重きをおき，3 章で述べた従来のセキュリティエンジニアリングが重視してきた部分に対してはどのように実施するのかについて，不明確である．STPA-Sec はミッション・ビジネス運用とシステム脆弱性に焦点を当てて，ハザード分析を行うものが公開されているだけで（図 6），システマチックな脅威分析を実施する手順や事例は公開されていない．脅威分析はハザード分析に対応するセキュリティエンジニアリングにおいて重要な分析であり，ハザード分析 STAMP/STPA への脅威分析の追加は必須である

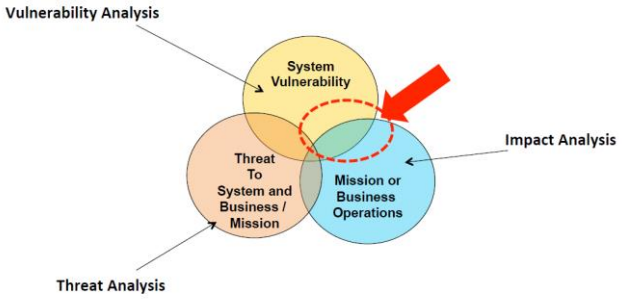

図 6. 現在の STPA-Sec の焦点

事例[6]をみても非安全なコントロールアクションに対して，セキュリティ制約をどうやって，導き出すかの手法が提示されていない．また， STPA 分析手順の Step2（図 5）に対応する段階では非安全なハザード要因を特定するが，現在の STPA-Sec では要因のシナリオを導き出す際にセキュリティ要因の必要十分性を説明できないのが現状である．セキュリティ要因の洗い出しは，ヒントワードに対して青字で記載した事項の追加により（図 7）対処することになっているが，これらのヒントワードはハザード要因分析のヒントワードに部分的，悪い形状の情報オペレーションを追加しただけである．なぜ，セキュリティ要因の洗い出しのこれらの追加がなされたのかの説明がなされていない．本来，セキュリティ誘発要因（SCF）には，悪意ある者の攻撃に基づいた要因分析結果が必要である．

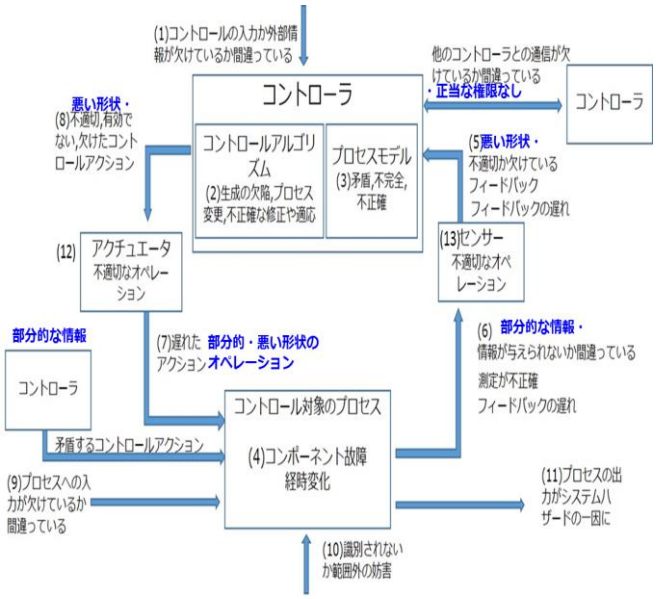

図 7. STPA-Sec におけるセキュリティ要因の洗い出し

4.2 脅威分析の追加方式

そこで，SCF の脅威洗い出しができる STPA-Sec とは異なる新たなヒントワードを定義し，脅威分析を STPA-Sec の手順に追加する方式を提案する．攻撃者の意図を踏まえたセキュリティ本来の視点から，より網羅性，実用性をもったセキュリティシナリオが作成できる手順に改善するためである．具体的には代表的な脅威分析モデルである Microsoft 社の STRIDE と「STRIDE に基づく脅威分類である」Threat Tree を手順に組み入れることとした．

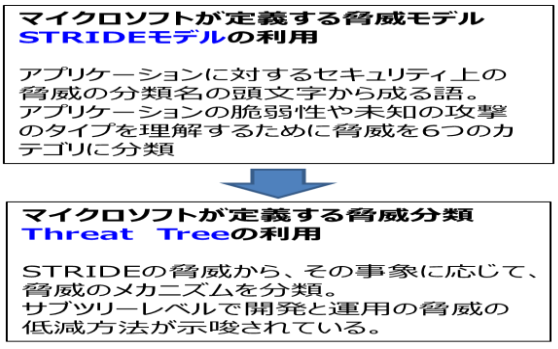

図 8. STRIDE モデルと Threat Tree

脅威分析を組み込む個所は，非安全なコントロールアクションに対するセキュリティ制約，セキュリティ要因を洗い出す STPA-Sec（Step2）の段階とする．

脅威シナリオを作成し，設計への推奨事項を示すための具体的な手順案（図 9）を以下に示す．

①非安全なコントロールアクション（UCA）ごとに STRIDE 脅威モデルで作成したヒントワードを適用することにより，

内部犯行か外部攻撃別に脅威を想定し，STRIDE 別に分類した UCA のタイプを特定する．
②特定した UCA のタイプが Threat Tree 分類のどこに相当するかを確認することにより， SCF として UCA の要因を突き止める．
③Threat Tree 分類の起こりうる脅威の説明を当該 SCF 状況に即して検討することにより，脅威シナリオを構築する．
③影響範囲が全体か部分的か一時的か恒久的かの 4 つに分けてインパクトを評価する方法である D4 インパクトなどの評価基準を利用することにより，脅威のインパクトを評価する．
④Threat Tree 分類の開発と運用による対策を利用することにより，設計の推奨事項を当該事例に即して作成する．

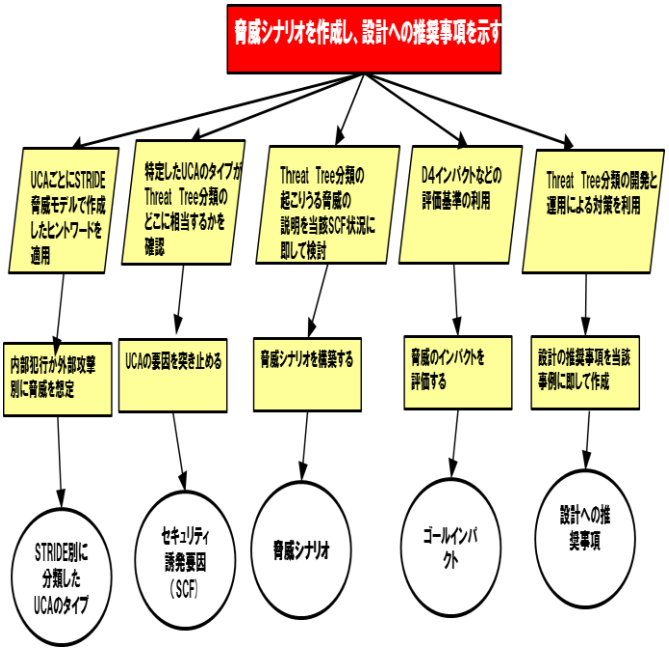

図 9. 網羅性，実用性をもったセキュリティシナリオ作成

上記手順の「①UCA ごとに STRIDE 脅威モデルで作成したヒントワードの適用」について，以下に詳細を説明する．非安全なコントロールアクションごとに脅威分析をして脅威シナリオを作るために STRIDE（図 10）を適用する．

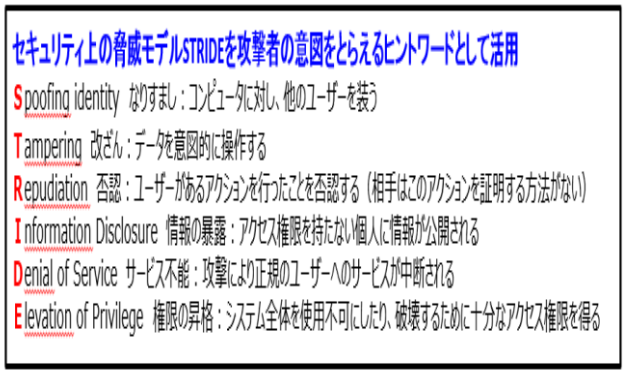

図 10. STRIDE の内容

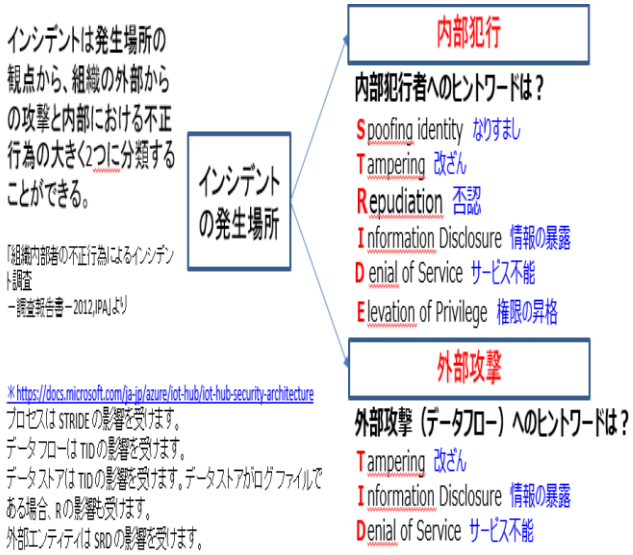

図 11. 内部犯行と外部攻撃による STRIDE の分類

STRIDE に当てはめてセキュリティ要因を分析する際に，ない．インシデントは発生場所の観点から，組織の外部からの攻撃と内部における不正行為の大きく 2 つに分類することができる[22]．マイクロソフト社によると STRIDE の 6 つの分類のうち，プロセスは 6 つの STRIDE の影響を受け，データ フローは STRIDE のうちの T と I と D（TID）の 3 つの影響のみを受ける．データストアも TID の 3 つの影響のみを受ける．なお，データストアがログファイルである場合，R の影響も受け，外部エンティティは 3 つの S と R と D（SRD）の影響のみを受けるとされている．この影響度合いを鑑みると，SCF を洗い出すヒントワードとして内部犯行（人）の場合と外部攻撃（データフロー）の場合に分類可能となる（図 11）．これらのセキュリティ要因は脅威モデルに基づき，抽象性はもつものの，必要十分性を明示することができる．

これらのヒントワードで要因を洗い出したうえで，さらに STRIDE のモデルとリンクした Threat Tree 脅威分類を利用し，脅威の発生するメカニズムに基づき，セキュリティ要因と対応する対策をシステムチックに導き出すことができる（図 12）．これらの手順を踏むことにより，セキュリティ対策の設計への推奨事項はアドホックではなく，脅威の発生するメカニズムを網羅的，かつ実用的におさえたものとなる．

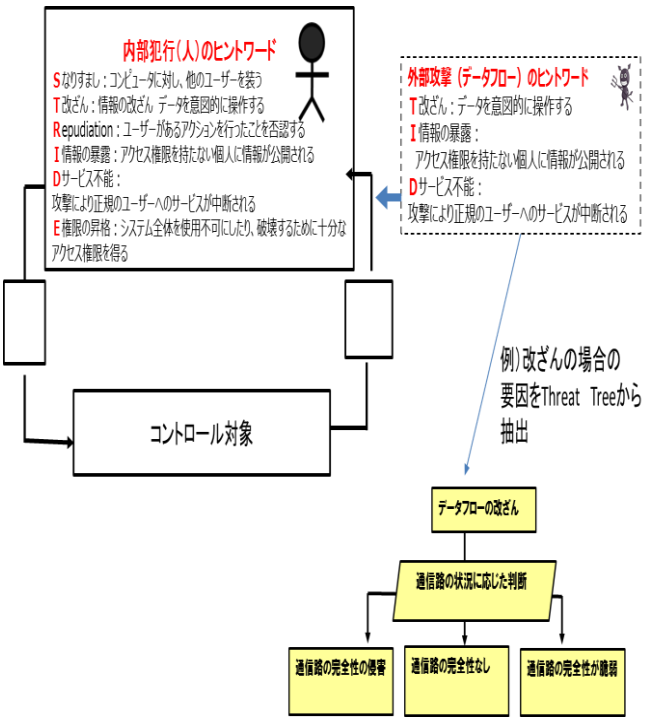

図 12. セキュリティ誘発要因（SCF）を洗い出すヒント

5. 脅威分析の追加方式による事例

本章では，脅威分析の追加方式に基づく航空機データ通信システム（図 13）の事例[23]を紹介する．

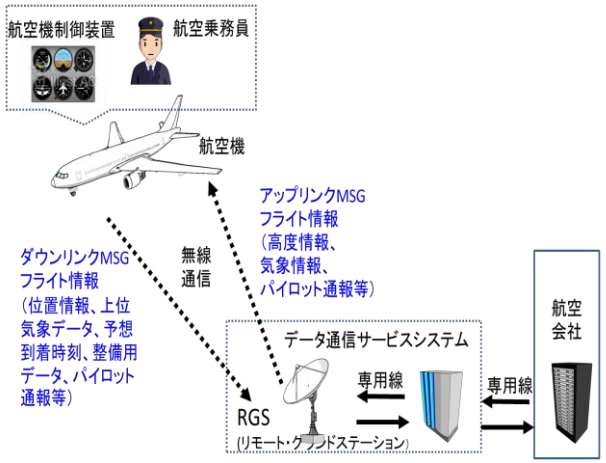

図 13. 事例のシステム構成

航空機データ通信システムでは，航空機の運航に必要な様々なデータを，航空機と地上（航空会社のシステム）の間で相互に通信する．地上から航空機への通信（アップリンク）では，性能情報，気象情報等があり，航空機からの通信（ダウンリンク）では出発・到着時刻，航空機の位置，整備用データ等がある．なお，これらの通信は航空機上や地上のコンピュータにより，ほとんどが自動化されている．今回は航空機の安全に対する影響が大きい飛行中を分析対象とする．

STPA-Sec の手順に基づき，以下のように分析を実施した．

1) 問題の定義と組み立て

今日，フライトオペレーションにはメンテナンスを含めサイバーアタックの脅威が高まっている．テロも含む多くの攻撃要素がありうるため安心安全なフライトは最重要であり，安全をセキュリティの視点からも分析可能であると考えられるためである．

ミッションは「乗客の生命、健康、財産にかかわる重大な事故を防ぎ、エアラインに航空産業における信用の向上につながるサービスを提供する」ことである．

主要なステークホルダーは，航空会社，データ通信システム，運航乗務員，航空機である．

システムの目的とゴールは「セキュアで安全なフライトを提供するための民間航空システム，航空会社のミッションをサポートするためのフライトオペレーション」である．

2) 受け入れられない損失とシステムハザード

情報セキュリティ上の機密性，完全性と可用性に加え，健康・生命への影響を与える安全の視点をもって受け入れられない損失とシステムハザードを考えると表 1 の項目があげられる．

表 1. 受け入れられない損失／アクシデント

#ID	受け入れられない損失／アクシデント
A1	乗客の生命にかかわる重大な事故
A2	個人情報（機内端末で決済するときに使用する乗客のクレジットカード情報など）の漏えい
A3	航空産業における信用の損失
A4	遅延

システムハザードとそれに対するシステム制約をあげると表 2 になり，受け入れられない損失／アクシデントとハザードの関係を整理したものが表 3 になる．

表 2. システムハザードとシステム制約

システムハザード	システム制約
H1-1:運航乗務員が航空機の制御操作をできない	SC1-1:運航乗務員は航空制御装置を正しく制御できなければならない
H1-2:航空機制御装置に不正アクセスがなされる	SC1-2:航空機制御装置は不正アクセスされてはならない
H1-3:航空機が特定のルートを外れる	SC1-3:航空機はフライトプランで特定されたルートを外れてはならない
H1-4:航空機が最低高度/最高高度を侵害する	SC1-4:航空機は事前に定められた最低/最高高度を侵害してはならない
H1-5:航空機が他の航空機との最低距離を侵害する	SC1-5:航空機は他の航空機との最低距離を侵害してはならない
H1-6:航空機が定刻通りに運航できない	SC1-6:航空機は的確に運航されなければならない

表 3.　受け入れられない損失／アクシデントとハザード

	A1：乗客の生命にかかわる重大な事故	A2：個人情報の漏えい	A3：航空産業における信用の損失	A4：遅延
H1-1：運航乗務員が航空機の制御操作をできない	×		×	
H1-2：航空機制御装置に不正アクセスがなされる	×	×	×	×
H1-3：航空機が特定のルートを外れる	×		×	
H1-4：航空機が最低高度/最高高度を侵害する	×		×	
H1-5：航空機が他の航空機との最低距離を侵害する	×		×	×
H1-6：航空機が定刻通りに運航できない			×	×

主要なステークホルダーをもとに相互のコントロール関係を図示したものが図 14 である．このコントロールストラクチャーにおいて，運航乗務員から航空機制御装置へのコントロールアクションが「CA1：航空機の制御装置にフライト情報の指示を与える」であり，航空機制御装置から航空機へのコントロールアクションが「CA2：制御装置が航空機を制御する」である．

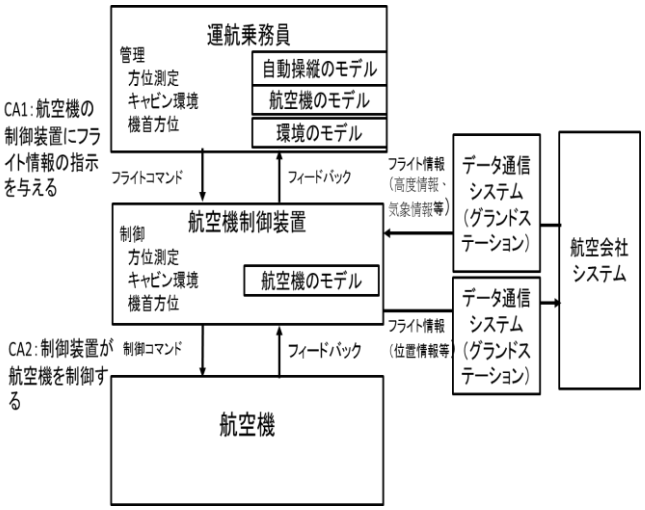

図 14.　機能コントロールストラクチャーの作成

STPA-Sec（STEP1）ではまず，導出したコントロールアクションごとに，４つのガイドワードに従い，非安全なコントロールアクションを洗い出し，どのシステム制約に違反するのかを示す（表 4）．

表 4.　STPA-Sec（STEP1）

コントロールアクション	非安全なコントロールアクション			
	与えられないとハザード	与えられるとハザード	早すぎ、遅すぎ、誤順序でハザード	早すぎる停止、長すぎる適用でハザード
CA1：航空機の制御装置にフライト情報の指示を与える	(UCA1-N)フライト情報が航空機制御装置に指示されない場合、ハザードに至る[SC1-1違反]	(UCA1-P)誤ったフライト情報が航空機制御装置に指示された場合、ハザードに至る[SC1-1、SC1-2、SC1-3、SC1-4、SC1-5違反]	(UCA1-T)航空機制御装置へのフライト情報指示が遅すぎた場合、ハザードに至る[SC1-6違反]	N/A
CA2：制御装置が航空機を制御する	(UCA2-N)航空機に飛行制御が与えられないときに、ハザードに至る[SC1-3、SC1-4、SC1-5、SC1-6違反]	(UCA2-P)航空機が誤った飛行制御を与えられたときに、ハザードに至る[SC1-3、SC1-4、SC1-5、SC1-6違反]	(UCA2-T)航空機に飛行制御が遅すぎるときに、ハザードに至る[SC1-6違反]	N/A

現在の STPA-Sec はこの次にセキュリティ制約を導き出す手順となる．MIT の事例ではセキュリティ制約をどうやって，導き出すかの手法が定まっていないが，脅威分析の追加方式ではコントロールストラクチャーをもとにコントローラーが内部犯行と外部攻撃に相当するのか分けて，網羅的に脅威を導出する．例えば，運航乗務員が攻撃者となった場合の脅威は内部犯行であり，STRIDE の脅威ごとにセキュリティ誘発要因を洗い出す（図 15）．

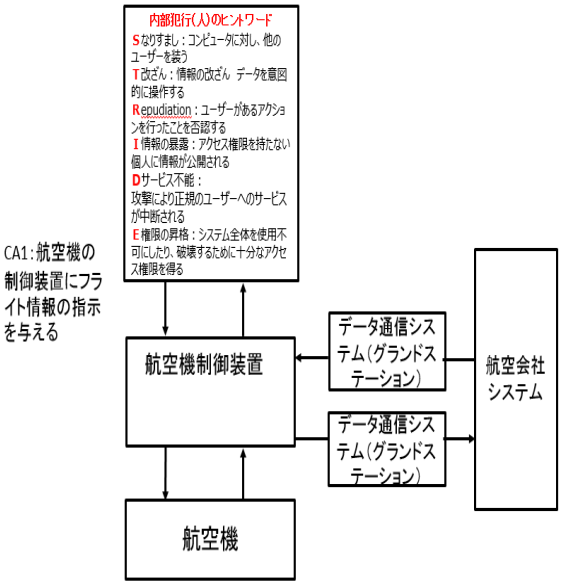

図 15.　内部犯行による STRIDE の分類

データ通信システムのグランドステーションからのインプットからの脅威を洗い出す場合は外部攻撃として TID の脅威ごとにセキュリティ誘発要因を洗い出す（図 16）．

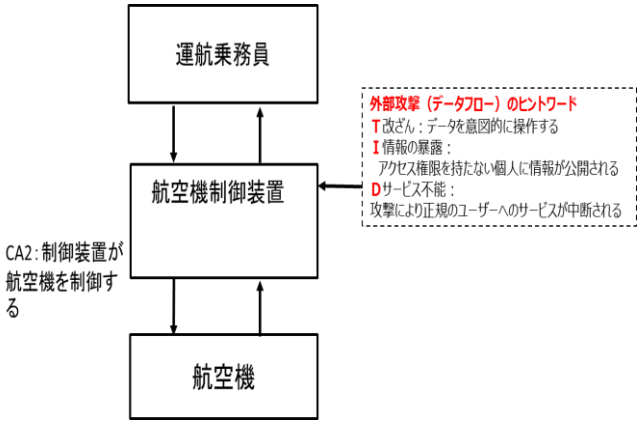

図 16.　外部攻撃による STRIDE の分類

例えば，洗い出したセキュリティ誘発要因 1 つとして T 改ざんをもとに脅威分類 Threat Tree を確認すると，この場合の脅威は Threat Tree の中の「メッセージの改ざん」の「メッセージの完全性なし」が想定できる．メッセージの改ざんサブツリー（表 5）を参照すると，「改ざんからメッセージの完全性を保護するものがない」説明と開発の低減策には「完全性の管理を加える」，運用の低減策には「データフ

ローにかなり依存する．トンネリングが脅威の一部に対応する可能性はある．しかしトンネリングは実際にはチャンネルの安全性を提供するものである.」という低減策が得られる．このように STRIDE にリンクした脅威メカニズムを示した分類 Threat Tree [21]を利用することで, STPA で攻撃の起こりうる個所に対して洗い出した脅威ごとに，開発と運用による脅威の低減策が得られる．

表 5．Threat Tree の「メッセージの改ざんサブツリー」

ツリーノード（TREE NODE）	説明（EXPLANATION）	開発低減策（DEVELOPER MITIGATION）	運用低減策（OPERATIONAL MITIGATION）
メッセージの完全性なし（No message integrity）	改ざんからのメッセージの完全性を保護するものがない（No Message integrity）	完全性の管理を加える	データフローにかなり依存する. トンネリングが脅威の一部に対抗し得る可能性はある. しかしトンネリングは実際にはチャンネルの完全性を提供するものである.
メッセージの弱い完全性（Weak Message	MDSのような弱いアルゴリズム	より良いアルゴリズムを利用する	上と同様
メッセージの弱い鍵管理（Weak Key managament）	弱い鍵管理はメッセージの問題につながる可能性がある	より良い鍵管理を用いる	上と同様

6. 提案方式に関する考察

脅威分析のための分類・手法は提案した STRIDE，Treat Tree 以外にも, 共通攻撃パターン列挙分類 CAPEC（Common Attack Pattern Enumeration and Classification）[24]や共通脆弱性タイプ一覧 CWE（Common Weakness Enumeration）[25]がある．これらは非常に詳細な分類である．STPA は詳細な設計段階ではなく，要求を抽出し必要な機能を見出す段階に用いる手法であるため，ヒントワードとして用いるためには向かないと判断した．

7. おわりに

本論文ではセキュリティ本来の視点からより網羅性，実用性をもったセキュリティシナリオを導出するため，STAMP/STPA-Sec とは異なるヒントワードを用い、手順に STRIDE と Threat Tree を用いた脅威分析を追加する方式を提案した. この方式は安全解析手法 STAMP/STPA の拡張提案である．本論文により，ハザード分析のためセーフティ分野で用いられてきた STAMP/STPA が 脅威分析を伴うセキュリティリスクの分析手法として利用できることを示した．

今後の課題としては，事例の詳細化と提案方式のできるだけ定量的な検証，セーフティセキュリティの統合化の検討が必要である．セキュリティサイドの人にとって安全を含めて分析できるセキュリティ要求分析手法であり，セキュリティに縁のなかったセーフティサイドの人でも確実に分析できる手法として本提案を確立していきたい．

参考文献

1) NISC,安全な IoT システムのためのセキュリティに関する一般的枠組

2) Nancy G. Leveson, Engineering a Safer World, Systems Thinking Applied to Safety ,2012

3) ナンシー・G・レブソン，セーフウェア，安全・安心なシステムとソフトウェアを目指して

4) IPA，はじめての STAMP/STPA,2016

5) IPA，はじめての STAMP/STPA（実践編）, 2017

6) William Young, Nancy Leveson. Systems Thinking for Safety and Security, Proceedings of the 29th Annual Computer Security Applications Conference （ACSAC 2013）, pp.1-8 （2013）.

7) William Young, Reed Porada, System-Theoretic Process Analysis for Security（STPA-SEC）:Cyber Security and STPA, 2017 STAMP Conference

8) Ivo Friedberg, Kieran, Paul Smith, David Laverty and Sakir Sezer. STPA-SafeSec: Safety and security analysis for cyber-physical systems, Journal of Information Security and Applications, Volume 34, Part 2, pp.183-196 （2017）.

9) NISC ,ww.nisc.go.jp/conference/seisaku/dai15/pdf/15siryou02.pdf

10) Schneier, B, Attack Trees. Dr. Dobb's Journal of Software Tools 24（12） （1999） 21–29

11) Barbara Kordy, Sjouke Mauw, Saša Radomirović, Patrick Schweitzer, Foundations of Attack–Defense Trees

12) Sindre, G. and Opdahl. L. A. : Eliciting security requirements with misuse cases, Requirements Engineering, Vol.10, No.1, pp. 34-44（2005）.

13) Mouratidis, H. : Secure Tropos homepage, （online）, available from <http://www.securetropos.org/>.

14) Liu, L., Yu, E. and Mylopolos, J. : Security and Privacy Requirements Analysis within a Social Setting, Proc. IEEE International Conference on Requirements Engineering （RE 2003）, pp.151-161（2003）.

15) Lin, L. Nuseibeh, B. Ince, D. et al. : Introducing Abuse Frames for Analysing Security Requirements, Proc. IEEE International Conference on Requirements Engineering （RE 2003）, pp.371-372 （2003）.

16) 金子朋子，山本修一郎，田中英彦：アクタ関係表に基づくセキュリティ要求分析手法（SARM）を用いたスパイラルレビューの提案,情報処理学会論文誌 52 巻 9 号（2011）

17) Kaneko,T.,Yamamoto, S. and Tanaka, H.: Specification of Whole Steps for the Security Requirements Analysis Method （SARM）- From Requirement Analysis to Countermeasure Decision -,Promac2011

18) Mead, N. R., Hough, E. and Stehney, T. :Sequrity Qualty Reqirements Engineering（SQUARE） Methodology（CMU/SEI-2005-TR-009）, www.sei.cmu.edu/publications/documents/05.reports/05tr009.html

19) Mead, N. R, 吉岡信和: SQUARE ではじめるセキュリティ要求工学,「情報処理」Vol.50 No.3（社団法人情報処理学会，2009 年 3 月発行）

20) Steve Lipner ,Michael Howard,:信頼できるコンピューティングのセキュリティ開発ライフサイクル, https://msdn.microsoft.com/ja-jp/library/ms995349.aspx

21) Adam shostack, Threat Modeling: Designing for Security, Wiley 2014

22) IPA,組織内部者の不正行為によるインシデント調査－調査報告書－,2012

23) 航空機データ通信システム事例, http://www.avicom.co.jp/services/data_link/example.html

24) IPA，CAPEC，http://capec.mitre.org/

25) IPA，CWE，www.ipa.go.jp/files/000024379.pdf