ゼロトラストアーキテクチャ

適用方針

2022 年（令和 4 年）6 月 30 日

デジタル庁

〔標準ガイドライン群ＩＤ〕
DS-210

〔キーワード〕
ゼロトラスト、ゼロトラストアーキテクチャ、

〔概要〕
政府情報システムのシステム方式について、より堅牢なシステム構築の観点からゼロトラストアーキテクチャの適用方針を示す。

改定履歴

改定年月日	改定箇所	改定内容
2022年6月30日		初版決定

目次

１　はじめに

１．１　背景と目的

政府情報システムにおけるサイバーセキュリティは、閣議決定された「デジタル社会の実現に向けた重点計画[1]」（以下、「重点計画」という。）において、利便性の向上と両立させることとしている。また、重点計画の「包括的データ戦略」では、新たな価値を創出する社会を実現するための方向性が示されている。そのために「サイバーセキュリティ戦略[2]」では、不確実性が絶えず変容し、かつ増大するサイバー空間の「自由、公正かつ安全」を確保し、さらに確保すべき価値の不変性に繋げるために、自らも常に変化を重ねていく必要があると示している。

今後、政府情報システムにおいては、標準の策定や共通機能の整備が進むと、重点計画の「国の情報システムを整備する際に留意すべき事項」の共通基盤の活用で言及されているように、クラウド・バイ・デフォルト原則に従い、クラウドサービスの利用を原則とした業務環境が目指されている。またテレワークの推進によりこの流れはより強化され、イントラネット外の業務環境を前提としたサイバーセキュリティ対策を実施する必要がある。

このような業務環境の変化により増大する脅威や攻撃経路は、従来の対策を凌駕し、深刻な被害をもたらす。イントラネット内での業務に加え、インターネット上での業務が標準になった場合でも、サイバー攻撃からリソースを保護しなければならない。近年の高度化したサイバー攻撃を完全に予防・防御することは、従来の考え方と対策だけでは困難になってきた。そのため、セキュリティインシデントの検知と対応、そして政府情報システムの復旧が重要になってきている。イントラネット内に限定されない業務環境におけるサイバーセキュリティを確保するために、従来のネットワーク防御を中心にした境界型セキュリティから考え方を大幅に拡張することが求められている。

そのため、この拡張の実態であり、境界型セキュリティを拡張した考え方である「ゼロトラストアーキテクチャ」を適用するための基本的な方針および留意事項を示す。各府省の政府情報システムのセキュリティ対策を検討する際には当該方針および留意事項を参照することを推奨する。

１．２　適用対象

本文書は、政府情報システムを適用対象として想定している。なお、本文書

[1] https://www.digital.go.jp/assets/contents/node/basic_page/field_ref_resources/5ecac8cc-50f1-4168-b989-2bcaabffe870/d130556b/20220607_policies_priority_outline_05.pdf

[2] https://www.nisc.go.jp/pdf/policy/kihon-s/cs-senryaku2021.pdf

はゼロトラストアーキテクチャへの理解を深める参考文書であり、適用の遵守を求めるものではない。

１．３　位置づけ

本文書は、標準ガイドライン群の一つとして位置づけられる。

１．４　用語

本文書において使用する用語は、表１-1 及び本文書に別段の定めがある場合を除くほか、標準ガイドライン群用語集の例による。その他専門的な用語については、民間の用語定義を参照されたい。

表　1－1　用語の定義

用語	意味
境界型セキュリティ[3]	従来のセキュリティの基本的な考え方であり、境界で内側と外側を遮断して外部からの攻撃や内部からの情報流出の防止を試みる。境界型セキュリティは「信頼できないもの」が内部に入り込まない、また内部には「信頼できるもの」のみが存在することが前提となる。セキュアなネットワークの構築が中心となる。
アクセス制御ポリシー	ユーザ属性情報、OS やパッチバージョンなどのデバイス属性情報といった情報から、アクセスが許可される条件を満たすかどうかを評価するルール。複数のポリシーを組み合わせて評価することも可能。
アクセス制御の施行	何らかの機構が、一定のルールやポリシーに従って評価されたアクセス可否の結果を実際のアクセスに対して施行すること。
ゼロトラスト	境界の内部が侵害されることも想定したうえで、情報システムおよびサービスの要求ごとに適切かつ必要最小の権限でのアクセス制御を行う際に、不確実性を最小限に抑えるように設計された概念。
ゼロトラストアーキテクチャ	ゼロトラストアーキテクチャは、ゼロトラストの概念を利用し、クラウド活用や働き方の多様化に対応しながら、政府情報システムのセキュリティリスクを最小化するための論理的構造的な考え方。

[3] https://cio.go.jp/sites/default/files/uploads/documents/dp2020_03.pdf

用語	意味
常時リスク診断情報	各関係組織のシステムにおけるサイバーセキュリティリスクについて常時かつ継続的に状況を把握し、必要に応じて各組織と連携してリスク低減活動を実施するための、情報収集した情報。政府情報システムにおいてはCRSA（Continuous Risk Scoring and Action）により収集された情報が該当する。
耐タンパー性を有するモジュール	暗号化・復号・署名生成のための鍵をはじめとする秘密情報や秘密情報の処理メカニズムを外部から不当に観測・改変することや秘密情報を処理するメカニズムを不当に改変することが極めて困難であるように意図して作られたハードウェアやソフトウェアのモジュール[4]。

１.５　ゼロトラストアーキテクチャとは

1)ゼロトラストアーキテクチャの概要

ゼロトラストアーキテクチャは、クラウド活用や働き方の多様化で増大する脅威に適合するために、政府情報システムの内部における攻撃者の自由な行動を阻害しようとするセキュリティ対策の考え方である。具体的には、特定の業務フロー内で、あるリソースから別のリソースへのアクセスが最小権限の原則を満たすよう、業務フローを取り巻く環境の情報を活用し、事前に定められたアクセス制御のルールによって評価され、その結果に従うアクセス制御が施行されるといった一連の手続きを踏む考え方である。この考え方は政府情報システムの導入や運用といった特定のフェーズだけに限定されるものではなく、中長期的な政府情報システムに係るライフサイクル全体にわたって適用されるものである。したがってゼロトラストアーキテクチャは特定の実装やソリューションを指すものではない。図 １－１は、ゼロトラストアーキテクチャの概念図である。

[4] https://www.ipa.go.jp/security/enc/CRYPTREC/fy15/documents/INSTAC_rep.pdf

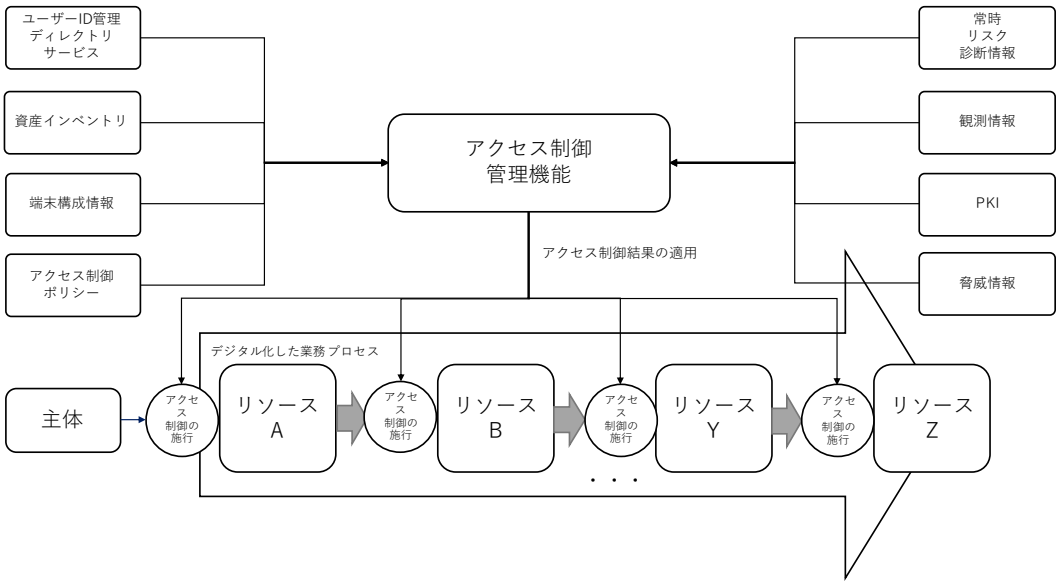

図 1-1 ゼロトラストアーキテクチャ概念図

デジタル化された業務プロセスにおいては、特定のリソースがアクセス元として、別のリソース（アクセス先）にアクセスする。表 1-2 では、従業員（人）がデバイスを用いてアプリケーションに保存されたデータにアクセスする業務プロセスと、主体がサーバであり複数のアプリケーションを介して特定のリソースにアクセスする業務プロセスをとりあげている。これらのアクセス内で、どのようなリソースが存在しうるかを簡易的に一覧化している。

リソース間の関係性は業務プロセス内で変化し、あるアクセスではアクセス先だったリソースが、別のアクセスではアクセス元になりうる。例えば、リソース A のアクセス先であったリソース B は、リソース Y との関係性ではアクセス元となる。そして、それぞれのアクセスにおいて、リソースの状態やアクセス時の文脈に応じたアクセス制御が適用される。

表 1-2 リソース例

主体	リソース A	リソース B	リソース X	リソース Z
人	デバイス	ネットワーク	アプリケーション X	アプリケーション X 内のデータ
サーバ	アプリケーション A	アプリケーション B	アプリケーション C	アプリケーション C の機能

2) 具体例

概念図をより詳細に説明するにあたって、「従業員が貸与された端末を使いファイルを参照する」業務フローを図 1－2に示す。

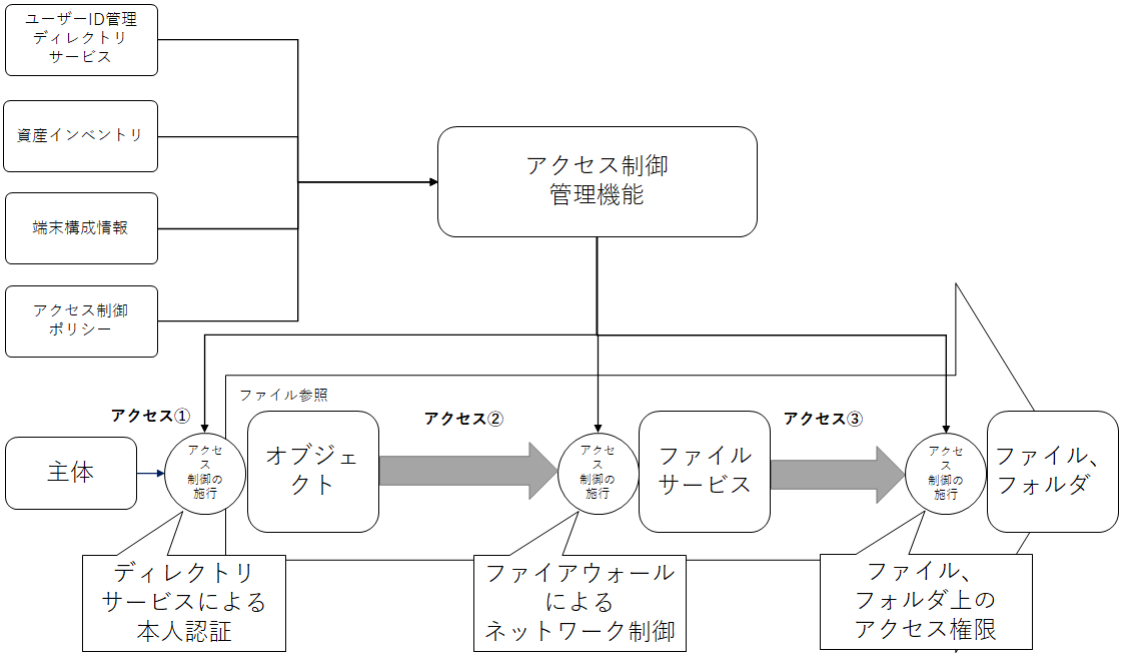

図 1－2　従業員が貸与された端末を使いファイルを参照する具体例

項番	業務内容	アクセス元	アクセス先	アクセス制御管理機能	アクセス制御の施行
①	端末上でのアカウントへのログイン	従業員（主体）	オブジェクト（ユーザ）	ディレクトリサービス	ディレクトリサービス
②	ファイルサーバへのネットワークアクセス	端末（主体）	ファイルサービス	ファイアウォール	ファイアウォール

③	ファイル参照	ユーザオブジェクト	ファイル	ファイルサービス	ファイル、フォルダ

表 1-3　従業員が貸与された端末を使いファイルを参照する具体例で発生するアクセス一覧

まず、従業員や端末といった物理的な主体がある。そしてデジタル上で表すリソースとして、ディレクトリサービス上のユーザオブジェクトやデバイスオブジェクトがあるとする。従業員や端末がそれらオブジェクトに認証を要求するとディレクトリサービスが課すアクセス制御ポリシーに評価され、認証されたのちに、必要最小限の権限がトークンやチケットという形で付与される（アクセス①)。

端末とファイルサービス間では、ネットワーク上のファイアウォールが送信元 IP アドレスや宛先ポート番号からアクセスを評価し、セッションを成立させる（アクセス②)。その後、ユーザオブジェクトがサービス内のファイルにアクセスする際は、ファイルやフォルダあるいはディレクトリ上のアクセス制御ポリシーがユーザオブジェクトあるいはデバイスオブジェクトに発行されたトークンを評価し、問題がなければファイルへの参照権限のみが付与される（アクセス③)。

3)境界型セキュリティとの関係

境界型セキュリティとは、境界で内側と外側を遮断して外部からの攻撃や内部からの情報流出の防止を試みる考え方であり、ネットワークを用いた実装が基本となる。境界型セキュリティは「信頼できないもの」が内部に入り込まない、また内部には「信頼できるもの」のみが存在することが前提となるが、標的型攻撃や内部犯行などの事例から本手法は限界があると言われ、ゼロトラストアーキテクチャとの比較対象とされることが多い。

しかし、本来、ゼロトラストアーキテクチャは境界型セキュリティとネットワークベースのセキュリティ対策を否定するものではない。図 1-3のように、パケットの送信元アドレスおよび宛先アドレスをオブジェクトと考え、特定のポートのみを許可するポリシーを適用していると考えれば、むしろゼロトラストアーキテクチャの考え方を正しく実現しているといえる。

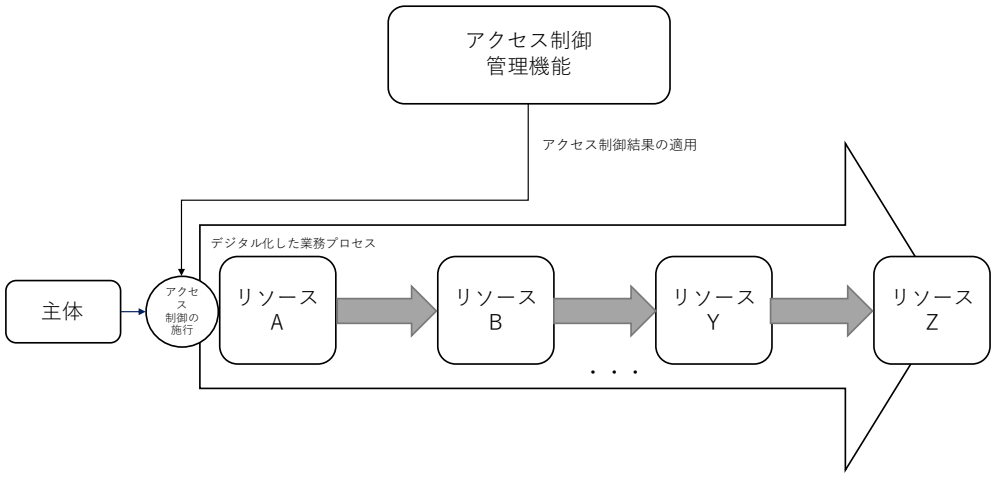

図 1-3 境界型セキュリティ概念図

一方、クラウド・バイ・デフォルト原則によって、今後インターネット上のクラウドサービス等の利活用が拡大すると見込まれ、オンプレミス環境とは異なる脅威や攻撃経路があると考えられる。そういった脅威に対し、ネットワークベースの境界型セキュリティだけではサイバーセキュリティ上の備えとしては不十分となる恐れがある。図 1-3の業務プロセスをクラウド上で行い、その際にアクセス制御をネットワーク境界だけで行う場合、より多くの脅威に晒されるインターネット空間上のリソース A からリソース Z までの一連の業務プロセスにおいて、アクセス制御が施行されるのはリソース A へのアクセス時のみになり、リソース B 以降には一切施行されないことになる。異なる業務環境に対して既存のセキュリティ対策を据え置くと、クラウド・バイ・デフォルト原則に期待される効率性の向上、セキュリティ水準の向上、技術革新対応の向上、柔軟性の向上、可用性の向上などのメリットを阻害しかねず、また、管理外のクラウドサービスの利用等といった抜け穴を増やす動機にも繋がる。業務環境の選択肢が増える中で、組織の目的とシステム戦略を踏まえて境界型セキュリティから変容させたものがゼロトラストアーキテクチャとなる。境界型セキュリティからゼロトラストアーキテクチャへの変容により、拡張された点について次に示す。

- **多様な業務環境への適用**
  - ✧ クラウド・バイ・デフォルト原則を前提にした環境であっても、既存のオンプレミスな境界型セキュリティな環境であっても、適用できる考え方になる。
- **複数の異なる情報を使ったアクセス制御**
  - ✧ リソース間同士のアクセス制御に細分化したことにより、ディレクトリサービスなどのネットワークパケット以外の情報を使ったアクセス制御が可能になる。
  - ✧ ネットワークベースの境界型セキュリティを追加のアクセス制御として扱える。
- **観測情報の入手の拡大**
  - ✧ 業務フローをリソースごとに区切り、それぞれにアクセス制御を施行するため、詳細かつ広範囲な観測データが入手できるようになる。
- **連携する外部システムの拡大**
  - ✧ 脅威情報や PKI など外部システムの属性情報と、アクセス制御管理機能が連携し情報の交換を想定している。
- **アクセス制御機能における評価と施行を分離**
  - ✧ アクセス制御におけるポリシーとの評価とアクセスへの施行を分離することで、柔軟な設計が可能となる。

2　適用方針

ゼロトラストアーキテクチャ適用において必要となる６つの適用方針を以下に示す。

1) リソースを識別し、特定できる状態にする

リソースを正確に特定できる状態でなければ、アクセス制御ポリシーの評価対象とすることはできない。そのため、リソースが識別できる状態で登録されていることが重要である。リソースは以下のものが考えられる。

- **アカウント**
  従業員に限らず、外部協力者など組織内業務に関連するあらゆるユーザのアカウントや、ディレクトリサービス上のオブジェクト。RPA などにも利用されるシステムアカウントも含まれる。
- **デバイス**
  スマートフォンやタブレットなど様々な種類の端末に加え、業務利用を許された個人端末が想定される。
- **サービス**
  オンプレミスなデータセンターやクラウドといった場所を問わず、ユーザが業務で利活用するアプリケーション等。
- **データ**
  デバイスやサービスに紐づくユーザが利活用するデータ。

リソースは組織内のみでなく、クラウド上や各従業員の自宅等様々な場所に存在することが想定される。また組織によっては個人所有のデバイスを業務利用として許可する事例もある。これらのリソースを全て手動で把握、管理することは現実的ではない。

そこで各種リソースを管理する資産管理ツールなどに登録し、ディレクトリサービス上のオブジェクトとして、ネットワークを介して管理することが考えられる。一例としては、デバイスを調達するサイトと端末管理サービスを紐付け、端末の初回起動時に自動的に資産として登録できるようにする仕組みが挙げられる。他にも、業務環境内のネットワークを対象に資産把握を目的としてスキャンをすることで、検出されたデバイスを一覧化することが考えられる。

2) 主体の身元確認・当人認証を実施する

利用者および端末などの物理的な主体は、システムを利用する際にはデジ

タルなリソースとして活動しなければならない。そのため、実際にその主体がリソースに正当に紐付けられているか、利用者に関しては、「行政手続におけるオンラインによる本人確認の手法に関するガイドライン[5]」に従って身元確認および当人認証によって確認しなければならない。

ア　身元確認

身元確認は、主体が管理対象であることを確認する行為を指す。利用者の場合は、登録する氏名・住所・生年月日等が正しいことを確認する。また、デバイスの場合は、購入時に入手したシリアル番号や割り当てたIPアドレスなどの管理上の情報と、実際に当該デバイス上で確認できる情報を照合することが考えられる。

確認内容および手法については、業務の重要性あるいは権限によって異なる。また、身元確認を実施するタイミングは、システムへの登録時だけでなく、忘却や紛失した際などをタイミングとした認証情報の再発行などで、主体とリソースの紐付けが要求されるすべての局面に適用するべきである。

イ　当人認証

当人認証はリソースの利用を試みる主体が身元確認によって紐付けられていることを知識（パスワード等）・所持（マイナンバーカード等）・生体（顔・指紋等）といった認証要素で確認することである。身元確認と同様に、当人認証も用途に応じた認証方法を適用するべきである。例えば、重要な権限を持つリソースへの当人認証の場合には、耐タンパー性[6]をもつハードウェアの利用を義務化することが考えられる。あるいは特定のネットワークアドレスからのみ認証を許可することや、特定の別の主体による承認を要求することも考えられる。昨今は多要素認証の設定を推奨されることが多くなったが、当人認証の手法に係る実装のレベルは「行政手続におけるオンラインによる本人確認の手法に関するガイドライン[7]」を参照して要件にあった認証方式を選択するべきである。

3)ネットワークを保護する

ゼロトラストアーキテクチャは、イントラネットを含めたネットワークを

[5] https://www.digital.go.jp/assets/contents/node/basic_page/field_ref_resources/e2a06143-ed29-4f1d-9c31-0f06fca67afc/f1be078e/20220422_resources_standard_guidelines_guideline_07.pdf

[6] https://www.ipa.go.jp/security/enc/CRYPTREC/fy15/documents/INSTAC_rep.pdf

[7] https://www.digital.go.jp/assets/contents/node/basic_page/field_ref_resources/e2a06143-ed29-4f1d-9c31-0f06fca67afc/f1be078e/20220422_resources_standard_guidelines_guideline_07.pdf

暗黙的に安全であるという前提を信用しない。そのため、ネットワークは通信経路の適切な暗号化によって安全性を確保しなければならない。具体的には、クラウド・バイ・デフォルト原則時に必ず利用される Web API の安全性を確保する HTTP や DNS の暗号化などが考えられる。この際には「電子政府推奨暗号リスト[8]」や「TLS 暗号設定ガイドライン～安全なウェブサイトのために（暗号設定対策編）～[9]」を参照し、適切な暗号技術が実装されていることを確認する。

4)リソースの状態を確認する

各種リソースは恒常的に安全とはいえない。適切に運用・保守されなければ、時間の経過とともに脆弱性が増え、攻撃可能な領域が広がる。また、設定ミス・構成の不備により脆弱性が生まれることも考えられる。そのため、各種リソースの状態や構成が安全か確認する必要がある。その確認作業はアクセス状態に関わらず常時実施されることが望ましい。何らかの手段によって、ログイン中のセッションを不正に利用されることもありうるからである。各種リソースの構成状態を確認するには、リソースの属性から総合的に診断することが重要である。具体的な属性例については以下に記す。

- ユーザの属性情報
- 当人認証に利用した認証要素
- デバイスの OS やミドルウェアのバージョン
- デバイスの構成情報
- サービスへの入力値
- アクセス時の位置情報

5)アクセス制御ポリシーで評価し、アクセス管理をする

各種リソース同士でアクセスを確立する際に、その可否を事前に定めたアクセス制御ポリシーを基にアクセス制御管理機能が評価し、その結果を施行できるようにしなければならない。また、ポリシーはその評価に応じて、別のアクセス制御ポリシーを呼び出す等、続くアクションを決定できるようにしなければならない。例えば、ある利用者のログインが普段とは大きく異なる位置情報から試行されている場合に追加の当人認証を求めるといったことが考えられる。また、アクセス元のデバイスの状態が身元確認済みのものであるかを確認することも考えられる。

---

[8] https://www.cryptrec.go.jp/list.html

[9] https://www.ipa.go.jp/security/vuln/ssl_crypt_config.html

将来的にポリシーそのものの内容を動的に変更する技術・ソリューションがリリースされることは否定できないが、一方、ポリシー自体は何かしらのガバナンス上のルールに基づいて決定されているものであるため、機械的に決定されたポリシーに対して説明責任を果たせる想定する必要がある。

6)リソースとアクセスを観測する

運用・保守をし、システムの信頼性を高めるうえで、リソースとアクセスのログの取得、アラートの通知など、政府情報システムを観測することが重要である。主な観測の目的は、次の事項の達成である。

- 導入したソリューション上での不具合やパフォーマンス上の問題追跡
- サブジェクト・オブジェクトの分析
- 変更内容などの追跡・管理
- 不審なアクセスの発見・調査
- 監査

全てを観測することはコストに上限があることから実現可能ではない。そのため、対象を観測することによって達成したい目的がなにか、明確にする必要がある。また、観測の要件を定め、リアルタイムで見るべき内容とリアルタイムではないが定常的な確認が求められる内容、ログを残しておくだけで充足する等、目的に応じた必要十分な観測をすることが求められる。

もし、不審なアクセスが認められた際は、組織内外の Security Operation Center（以下、「SOC」という。）チームと共同して対処することが求められる。その場合はログの連携をするか、個々のソリューション上のダッシュボードを通して双方が監視するなど、幾つかの手法が考えられる。

３　具体的な適用手順

３．１適用プロセス

ゼロトラストアーキテクチャは、中長期的に運営する政府情報システムを堅牢にするための考え方である。従って、「ゼロトラストアーキテクチャの適用」は一過性なプロジェクトではなく、ましてや単一のソリューションを導入したことで完了するものでもない。政府情報システムにおける特定の目標を特定の期間で実行するプロジェクト・サイクルを繰り返すことで成熟度をあげる行為が、ゼロトラストアーキテクチャの適用といえる。そのサイクルは図 ３−１の通りになる。なお、本章では図 ３−１に従い各プロセスを説明するが、プロセスはどこから開始しても良い。また、同様のプロセスが存在している場合には既存のプロセスで代替しても問題はない。

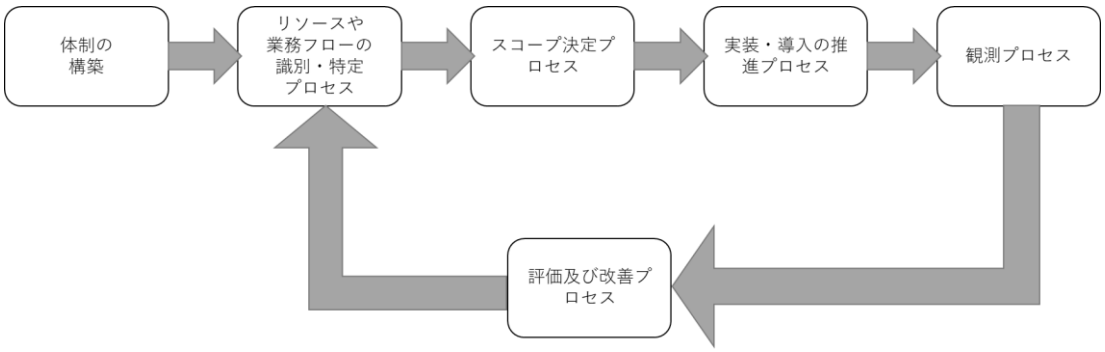

図 ３−１ゼロトラストアーキテクチャ適用プロセス

体制の構築

適切なゼロトラストアーキテクチャの考え方を政府情報システムへ適用するには、十分な体制が必要不可欠である。当該体制を実現するために招集する対象は、業務フローの流れなどについて最も理解が深いであろう業務担当者や開発者はもちろんのこと、監視業務等の運用や保守担当者も含まなければならない。業務、開発、運用は情報システムの運営という観点では排他的ではなく、相互に調整し最適解を見つけることが重要である。この最適解は、運用にあわせ業務フローを見直すことも含める。

また、それぞれのステークホルダー間の責任範囲を明確化し、責任分界点を設けることが望ましい。

1)リソースや業務フローの識別・特定プロセス

システム上のリソースと業務フローを識別・特定する。この際に、機密性・完全性・可用性などの観点から、リソースと業務フローに関する特性も分析の対象とすると、後続するプロセスでより活用しやすい。

なお、リソース識別時の粒度は、当該組織の過去の施策と現状、あるいは保護したい対象に応じて異なる。例えば、対象をデバイス上で直接動作するソフトウェアまでを対象とするか、あるいはサービスやソフトウェアが再帰的に依存するライブラリやモジュールまでを対象とするか、を判断する必要がある。

2)スコープの決定プロセス

リソースや業務フローに対する攻撃経路、脅威、リスクを、脅威モデリングなどによりスコープを特定する。先述したプロセスで分析した内容から影響度を算出し、それに応じた優先度や対応内容・粒度を検討する。

例えば、ユーザの身元確認・当人認証に高いリスクがあるのであれば、全利用者を対象にした多要素認証の必須化や、信頼できる ID プロバイダとのシングルサインオンの推進などが施策の候補に挙がる。あるいは、テレワークを前提とした中でネットワークに対するガバナンスを強化したい場合は、SDN（Software Defined Network）や SASE（Secure Access Service Edge）といったソリューションを全業務デバイスに導入するといった施策が考えられる。

3)実装・導入の推進プロセス

前項で定めたスコープをもとに施策を進めるが、変化する業務環境やセキュリティにあわせ、設計・開発・テスト・成果物のリリースといった一連の流れをより高頻度に行える状態にすべきである。

その場合、手動による変更は柔軟性が高いものの、頻繁な変更には向かない。構成をコード化し、変更のリリースを自動化することが望ましい。

4)観測プロセス

利用者のアカウントの動作、デバイスの状態、ネットワークなど、リソースに関する観測をする。観測は、ログの収集や分析、パフォーマンスの監視、アラート通知などが含まれる。SOC が利用する SIEM（Security Information and Event Management）を通して行われるケースもあれば、それぞれのログからアクセス制御ポリシーによって評価されることもありえる。どのように行

うかは、採用した技術やソリューションにより異なるが、監視結果に応じてどのようなアクションを起こすか、といったような観測の目的を明確化することが重要である。

5)評価及び改善プロセス

取り組んだ内容の有効性を評価し、次のサイクルに活用する洞察を得る。それによりゼロトラストアーキテクチャの考え方を適用したシステムの成熟度の向上が期待できる。

一方、評価方法については留意すべきである。増え続ける脆弱性や業務の変更を受け、システムの変更を頻繁に実施可能な状態にすることが望ましいのは先述した。その場合、従来の半年や年次の内部監査等の人的リソースに依存した評価方法は、システム変更の頻度・速度に比較すると十分ではないことが懸念される。そのため、CSPM（Cloud Security Posture Management）などの活用で、受動的な評価を自動化し、補強することが望ましい。独立組織による疑似攻撃による脆弱性の検証をおこなうレッドチーム演習のような能動的な評価も定期的に実施することが望ましい。

３．２適用における留意事項

ゼロトラストアーキテクチャを適用するには、適用プロセス内で実施する内容に留意しなければならない。

1)運用・保守体制を確保する

成熟度にもよるが、概念図における外部システムを含む各種コンポーネントやデータは、複数の組織外の担当者によって共有されることが予想される。運用・保守の対象はそれらコンポーネントやデータも含む。そのため、システムの運用担当者のリソース確保のため、ステークホルダーとの事前調整と合意が重要になる。当然、観測の結果、セキュリティインシデントなどが検出されるなどの場合も想定し、非定常的な事態を想定した連絡体制と協同体制も構築すべきである。

2)運用の設計と実装を初期段階から想定した適用プロセスを進める

上述した通り、ゼロトラストアーキテクチャの考え方が適用されたシステムでは、運用が複雑になることが予想される。ゼロトラストアーキテクチャ適用の目的はシステムの継続的な価値創出であり、既存実務とゼロトラストアーキテクチャ適用後の実務とのギャップを埋めることが重要である。政府情報システムが達成したい目標や、利用者の利便性を保つためにも、適用プ

ロセスの初期段階から、運用の設計・実装も同時に実行しなければならない。そうすることにより、修正の機会をより多く得られ、またスムーズな実稼働フェーズへの移行を実現できる。

3)アクセス制御の評価タイミングをアクセス要求時に限定しない

ゼロトラストアーキテクチャでは、アクセス制御ポリシーによる「評価」と、その評価結果を実際のアクセス制御として反映する「施行」が、それぞれ独立したアクションとなる。したがって、特定のタイミングでは評価を有効化するが施行はしない、といった状態が成立しうる。アクセス制御の施行前に業務への影響を検証する必要があれば、その状態を設定可能なソリューションを導入あるいは実装をするべきである。

また、アクセス制御の評価と施行の対象は、まだアクセスが確立されていないアクセス要求のみに限定されず、確立済みのアクセスも含まれる。ブラウザでの利用を考えると、セッション・ハイジャックは依然として存在する脅威になるからである。したがって、アクセスが確立された後も継続的にアクセス制御の対象になる。アクセス制御の手法は特性に応じて選択される。

アクセス評価の結果を施行する箇所が、アクセス元のリソースか、アクセス先のリソースか、あるいはアクセス元とアクセス先の中間にあるプロキシのようなリソースが実施するかについては、業務フローやシステム構成によって変わる。識別されたリソースや業務フローの特性に応じた構成を設計段階から検討すべきである。

例えば、複雑な業務フローと権限設定を前提とするアプリケーションがあるとする。この場合、利用者の権限は業務に依存する傾向があるため、権限管理をアプリケーション（アクセス先）に集約するケースも考えられる。つまり、アクセス制御の施行箇所がアクセス先になる。

一方、アクセス制御がアクセス元で施行されるケースも考えられる。例えば、モダンなネットワークソリューション（SASE, SDN 等）のデバイス・エージェントは、ネットワークトラフィックをデバイス上で評価し制御を施行するものもある。評価内容を示すポリシーは当該ソリューションの集中管理機能からエージェントに配布され、アクセス制御の評価・施行機能からは独立している。

また、アクセス制御の評価と試行が同じ箇所になることもありえる。例として、物理的なファイアウォール機器がある。これは、同一の機器にネットワークルールを設定でき、それに基づいてネットワーク制御が行われる。

アクセス制御の評価および施行方法については、業務におけるアクセス制御のあるべき姿、ステークホルダー間の責任分界点、導入予定あるいは導入

済みソリューション・実装の仕様にあわせて、構成を検討する必要がある。

4)技術標準による相互互換性を確保する

業務の現場において、様々な業務特有の独自性があることは否定できない。しかし、業務をシステムとして実装あるいはソリューションを導入する際には、既に標準化されたデータフォーマットや実装方式がある場合は、それらに従い、再定義することを避けるべきである。もし、どうしても独自に実装するのであれば、それによって発生するメリット・デメリットを注意深く検討するべきある。

技術標準に従わない場合、他のソリューションやサービス、あるいはそれらが提供するAPIとの連携が困難になる恐れがある。それはつまり、システム化の難易度および運用の複雑性が増し、人的リソースを含めた各種コストが上がることを意味する。

また、セキュリティ観点でも、例えば、暗号化や署名といったものを独自実装した場合、多人数によるアルゴリズムのチェックや、万が一それに問題があった際のパッチ作成が困難になり、結果的にセキュリティリスクが高まる恐れがある。パッチ作成が可能であったとしても、特別なものになるため、通常よりコストが上がることが想定され、また、ベンダーロックインのリスクも高まる。

本来の政府情報システムが提供したかった価値に制限がかかる可能性があるため、技術標準による相互互換性の確保は重要である。特定の用途で独自の実装要件が必要なケースはありえるが、同時にそういったケースは本ガイドラインを適用する政府情報システムおよび業務システムにおいては稀であると想定する。

5)利用者の問い合わせ対応を強化する

ゼロトラストアーキテクチャ概念図で示した通り、一連の業務フロー内で複数のアクセスが行われる。実際の業務を執行する利用者には、サーバ間のアクセスやデバイスからアプリケーションへのアクセスについては、ブラックボックス的になる。そのことから、利用者が想定していない動作に直面しても、どこで問題に陥っているかが認知されにくい。したがって、利用者の負担が増えると予想される。例えば、パッチを一定期間内に適用していないデバイスにリスクがあるとしてアクセスを拒否した場合、本人の直接的な操作に起因しないため、即座に本人がパッチ適用をするアクションに繋がらない恐れがある。本人が原因と対処方法を実行できることが理想的だが、サービス間がAPIを使う機械的なデータ連携をするフロー内で拒否あるいは意図し

ない挙動が起こった際は難しい。

そのため、エンドユーザから問い合わせへの一次対応を受け持つ部門が必要になることが予想されるが、その部門もユーザからの問い合わせから問題になったアクセス要求をトレース・特定し、適時、問題発生個所を所管する部門に連携することになると思われる。問い合わせ時の対応についても、設計段階から準備しておくべきである。担当部門の窓口を伝え利用者自身で問い合わせてもらうか、一次対応部門から連携するなどの方式が考えられる。

また、ユーザの画面に特定の問題を示すエラーコードや、トレースを容易にするためのアクセス要求識別子、簡易的なメッセージを表示するなど、エンドユーザの問い合わせ負荷や問い合わせ対応者の調査、対応者間の連携を円滑にする手法を想定すべきである。あるいは、その情報を自動提供できるような問い合わせ機能を提供することを検討するべきである。

4　参考文献

- デジタル社会の実現に向けた重点計画、令和４年６月 22 日、閣議決定
- サイバーセキュリティ戦略、令和３年９月 28 日、閣議決定
- デジタル・ガバメント推進標準ガイドライン、デジタル庁
- 政府情報システムにおけるゼロトラスト適用に向けた考え方、 ２０２２年６月、政府 CIO ポータル
- Zero Trust Architecture, August 2020, NIST Special Publication 800-207
- Zero trust architecture design principles, July 23 2021, NCSC
- 行政手続におけるオンラインによる本人確認の手法に関するガイドライン、平成３１年２月２５日、各府省情報化統括責任者（CIO）連絡会議決定
- オンラインサービスにおける身元確認に関する研究会、2020 年 4 月 17 日、経済産業省
- 電子政府推奨暗号リスト、2021 年４月１日、CRYPTREC
- 耐タンパー性調査研究委員会報告書、平成 15 年３月、IPA
- TLS 暗号設定ガイドライン～安全なウェブサイトのために（暗号設定対策編）～、2021 年 12 月７日、IPA
- Planning for a Zero Trust Architecture: A Planning Guide for Federal Administrators, May 6, 2022, NIST CSWP 20
- Moving the U.S. Government Toward Zero Trust Cybersecurity Principles, January 26, 2022, EXECUTIVE OFFICE OF THE PRESIDENT, OFFICE OF MANAGEMENT AND BUDGET M-22-09
- Shared Signals and Events – A Secure Webhooks Framework、OpenID Foundation