

ADOBE® AIR® のセキュリティ

法律上の注意

法律上の注意については、http://help.adobe.com/ja_JP/legalnotices/index.htmlを参照してください。

目次

AIR のセキュリティの概要

デスクトップアプリケーションのインストールとアップデート	1
モバイルアプリケーションのインストールとアップデート	3
Adobe AIR アップデート	4
コード署名	5
セキュリティサンドボックス	6
ファイルシステムへのアクセス	7
ネイティブプロセスとの通信	8
信頼されないコンテンツの安全な使用	8
Android デバイスのセキュリティ	9
iOS デバイスのセキュリティ	11
HTML セキュリティ	11
その他のセキュリティに関する考慮事項	12

AIR のセキュリティの概要

セキュリティは、アドビ システムズ社、ユーザー、システム管理者およびアプリケーション開発者にとって、重要な関心事の 1 つです。このような理由から、Adobe® AIR® には、ユーザーやアプリケーション開発者を保護するためのセキュリティ規則やコントロールのセットが用意されています。このホワイトペーパーでは、Adobe AIR アプリケーションを使用および開発する際のセキュリティに関する考慮事項について説明します。

AIR のセキュリティモデルは、Flash® Player で実行される SWF コンテンツやブラウザで実行される HTML コンテンツのセキュリティモデルを発展させたものですが、セキュリティコントラクトはブラウザのコンテンツに適用されるセキュリティコントラクトとは異なります。このコントラクトは、開発者に、ブラウザベースのアプリケーションには十分すぎるほど自由にリッチエクスペリエンスを実現できる豊富な機能を提供します。

AIR アプリケーションは、コンピューティングデバイス上の他のネイティブアプリケーションと同じオペレーティングシステムのセキュリティ制約で実行されます。一般的に、これらの制約では、ファイルの読み取りや書き込み、画面への描画、ネットワークとの通信など、オペレーティングシステム機能に広範囲にアクセスできます。ネイティブアプリケーションに適用されるオペレーティングシステムの制限（ユーザー固有の権限など）は、AIR アプリケーションにも同等に適用されません。

AIR アプリケーションは、コンパイル済みバイトコード（SWF コンテンツ）または変換されるスクリプト（JavaScript、HTML）のいずれかを使用して作成されているので、ランタイムでメモリを管理できます。これにより、メモリ管理に関連する脆弱性（バッファオーバーフローやメモリの破損など）によって AIR アプリケーションが影響を受ける可能性を最小限に抑えることができます。これらは、ネイティブコードで作成されたデスクトップアプリケーションに影響を与える最も一般的な脆弱性の一部です。

注意：このホワイトペーパーでは、Adobe AIR でのセキュリティに関連する事項について説明します。次の開発者向けドキュメントには、セキュリティで保護された AIR アプリケーションの開発に関する技術的な詳細や、AIR API を使用する際の考慮事項が示されています。

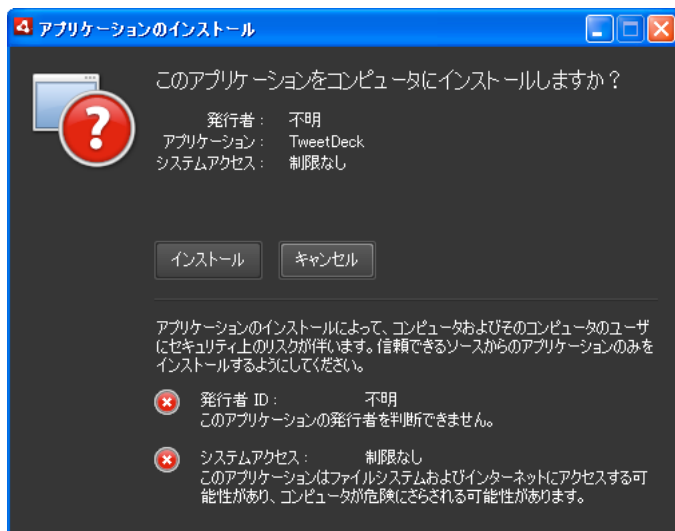
- ActionScript（Flash および Flex）開発者は、『ActionScript 3.0 開発ガイド』の [AIR のセキュリティ](#) を参照してください。
- Ajax 開発者は、『Adobe AIR 用 HTML 開発ガイド』の [AIR のセキュリティ](#) を参照してください。

デスクトップアプリケーションのインストールとアップデート

デスクトップ AIR アプリケーションは、拡張子が air の AIR インストーラーファイルを使用して配布されます。Adobe AIR がインストールされている環境で AIR ファイルを開く場合、ランタイムによってアプリケーションのインストールプロセスが制御および管理されます。

注意：開発者は、バージョン、アプリケーション名、発行者ソースを指定できますが、アプリケーションのインストールの最初のワークフロー自体は変更できません。Adobe AIR によって管理される安全で合理化された一貫性のあるインストール手順をすべてのアプリケーションで共有できるので、この制限はユーザーにとって有利になります。アプリケーションのカスタマイズが必要な場合は、アプリケーションを初めて実行するときに指定できます。

デフォルトのアプリケーションインストーラーによって、セキュリティ関連の情報がユーザーに示されます。AIR のインストール中に発行者名が表示されるのは、信頼できる証明書、またはインストールされているコンピューターで信頼されている証明書にチェーン化されている証明書を使用して AIR アプリケーションに署名している場合です。それ以外の場合、発行者名は「不明」として表示されます。これにより、ユーザーは、十分な情報に基づいてアプリケーションをインストールするかどうかを決定できます。



SWF ファイルで最初に Flash Player ブラウザープラグインをインストールする必要があるのと同様に、AIR アプリケーションでも最初にランタイムをユーザーのコンピューターにインストールする必要があります。

ランタイムは 2 つの方法でインストールできます。1 つはシームレスインストール機能を使用する方法で、もう 1 つは手動でインストールする方法です。

- シームレスインストール機能を使用すると、開発者は、Adobe AIR をインストールしていないユーザーに合理化されたインストールエクスペリエンスを提供できます。シームレスインストールでは、開発者は Web ページに SWF ファイルを埋め込み、その SWF ファイルでインストール対象の AIR アプリケーションの名前を示します。ユーザーが SWF ファイルをクリックしてアプリケーションをインストールするときに、SWF ファイルによってランタイムの有無が確認されます。ランタイムが検出されない場合、ランタイムはインストールされて、開発者のアプリケーションのインストールプロセスですぐに有効化されます。ユーザーには、インストールをキャンセルするためのオプションも提供されます。
- ユーザーが AIR ファイルをインストールする前にランタイムを手動でダウンロードしてインストールすることもできます。この場合、開発者は様々な方法（電子メールや Web サイト上の HTML リンクなど）で AIR ファイルを配布できます。AIR ファイルを開くと、ランタイムが有効化され、アプリケーションのインストール処理が開始されます。

AIR のセキュリティモデルでは、ユーザーが AIR アプリケーションをインストールするかどうかを決定できます。AIR インストーラーでは、ユーザーがこの信用判断をより簡単に行うことができるように、ネイティブアプリケーションのインストールテクノロジーにいくつかの改良が行われています。

- ランタイムは、AIR アプリケーションが Web ブラウザー内のリンクからインストールされる場合でも、すべてのオペレーティングシステムで一貫したインストールエクスペリエンスを提供します。ネイティブアプリケーションのインストールエクスペリエンスは、セキュリティ情報がすべて提供される場合、セキュリティ情報を提供するブラウザーまたは他のアプリケーションに依存します。
- AIR アプリケーションインストーラーはアプリケーションのソースを識別し（ソースを検証できない場合は、インストーラーが確認し）、ユーザーがインストールの続行を許可した場合にアプリケーションで利用可能になる権限に関する情報を提供します。

- ランタイムは、AIR アプリケーションのインストールプロセスを管理します。AIR アプリケーションは、ランタイムが使用するインストールプロセスを操作できません。

一般的に、信頼していないソースや、検証できないソースから提供されたアプリケーション（AIR アプリケーションを含む）はインストールしないでください。他のインストール可能なアプリケーションと同様に、ネイティブアプリケーションのセキュリティに関する立証責任は、AIR アプリケーションにも同等に当てはまります。

AIR 2 では拡張デスクトップ AIR アプリケーションがサポートされました。これらのアプリケーションはネイティブインストーラーファイルを使用してインストールされます。

- DMG ファイル（Mac OS 用）
- EXE ファイル（Windows 用）
- RPM または DEB パッケージ（Linux 用）

AIR アプリケーションのアップデート

ソフトウェアアップデートの開発と展開は、ネイティブコードアプリケーションの中でも最も大きなセキュリティ問題の 1 つです。インストール済みの AIR アプリケーションで、リモートの場所にアップデート AIR ファイルがあるかどうかを確認できます。アップデートが必要な場合、AIR ファイルがダウンロード、インストールされ、アプリケーションが再起動します。開発者向けドキュメントでは、この方法を使用して、新しい機能を提供するだけでなく、潜在的なセキュリティの脆弱性に対処するための詳細を説明しています。

AIR 2 では拡張デスクトップ AIR アプリケーションがサポートされました。これらのアプリケーションのインストールおよびアップデートには、次のネイティブインストーラーファイルが使用されます。

- DMG ファイル（Mac OS 用）
- EXE ファイル（Windows 用）
- RPM または DEB パッケージ（Linux 用）

ビルトイン AIR Updater クラスと AIR アップデートフレームワークでは、ネイティブインストーラーを使用してインストールされた AIR アプリケーションのアップデートはサポートされていません（アップデートをサポートするオープンソースプロジェクトを利用できます）。

関連項目

[RIASpace: Native Application Updater project](#)

AIR アプリケーションの削除

AIR アプリケーションを削除すると、アプリケーションディレクトリ内のすべてのファイルが削除されます。ただし、アプリケーションがアプリケーションディレクトリ以外のディレクトリに書き込んだファイルは削除されません。AIR アプリケーションを削除すると、AIR アプリケーションがアプリケーションディレクトリ以外のディレクトリに行った変更は元に戻りません。

モバイルアプリケーションのインストールとアップデート

モバイル AIR アプリケーションは、サポートされるプラットフォームのネイティブパッケージとして配布されます。Android のパッケージフォーマットは APK ファイルで、iOS のパッケージフォーマットは IPA ファイルです。ユーザーは、プラットフォームでサポートされている通常の方法でモバイル AIR アプリケーションのダウンロードおよびインストールを実行できます。例えば、Android では Market、iOS では App Store を使用します。AIR アプリケーションのインストールには、プラットフォームの他のアプリケーションと同じ制限が適用されます。

Android では、AIR ランタイムは個別にインストールされ、AIR for Android アプリケーションが起動されるたびに有効化されます。

iOS の各 AIR アプリケーションには必要な機能がすべて含まれているので、iPhone などの iOS デバイスでは、AIR ランタイムは個別にインストールされません。

一般的に、信頼していないソースや、検証できないソースから提供されたアプリケーション（AIR アプリケーションを含む）はインストールしないでください。他のインストール可能なアプリケーションと同様に、ネイティブアプリケーションのセキュリティに関する立証責任は、AIR アプリケーションにも同等に当てはまります。

モバイル AIR アプリケーションのアップデート

ソフトウェアアップデートの開発と展開は、ネイティブコードアプリケーションの中でも最も大きなセキュリティ問題の 1 つです。モバイルデバイス上の AIR アプリケーションでは、ネイティブプラットフォームのアップデートメカニズムを使用できます。Android で使用できるメカニズムは、Android Market です。iOS で使用できるメカニズムは、Apple iTunes App Store です。

Adobe AIR アップデート

アドビ システムズ社では、新機能や小さな問題に対する修正プログラムによって、Adobe AIR を定期的に更新しています。

デスクトップ AIR アップデート

デスクトップオペレーティングシステムでは、自動通知およびアップデート機能を使用すると、Adobe AIR のアップデートバージョンが利用可能になったときに、ユーザーに自動的に通知することができます。

Adobe AIR をアップデートすることで、Adobe AIR が適切に動作します。また、セキュリティに対する重要な変更が反映される場合があります。新しいバージョンが利用可能になった場合、特に、セキュリティアップデートについて言及されている場合は、必ず Adobe AIR の最新バージョンにアップデートすることをお勧めします。

デフォルトでは、AIR アプリケーションを起動したときに、ランタイムによって、アップデートが利用可能であるかどうかチェックされます。このチェックは、前回のアップデートチェックから 2 週間を越えると実行されます。アップデートが利用可能であれば、アップデートがバックグラウンドでダウンロードされます。

ユーザーは、AIR 設定マネージャーアプリケーションを使用して、自動アップデート機能を無効にできます。AIR 設定マネージャーアプリケーションは、

<http://airdownload.adobe.com/air/applications/SettingsManager/SettingsManager.air> からダウンロードできます。

Adobe AIR の通常のインストールプロセスでは、インストール環境の基本情報（オペレーティングシステムのバージョンや言語など）を送信するために、<http://airinstall.adobe.com> への接続が行われます。この情報は、1 回のインストールにつき 1 度だけ転送されます。この情報を使用することで、アドビ システムズ社では、インストールが成功したことを確認できます。個人を特定する情報を収集したり、転送したりしません。

モバイル AIR アップデート

Android では、AIR ランタイムのアップデートは Android Market で配布されます。

iOS では、ランタイムは各アプリケーションにバンドルされています。アプリケーション開発者は、アップデートされた SDK を使用してアプリケーションを再構築して、バグ修正プログラム、セキュリティアップデート、新しいランタイム機能を組み込むことができます。

コード署名

Adobe AIR では、すべての AIR アプリケーションが電子署名されている必要があります。コード署名は、ソフトウェアの整合性と発行者の ID を保証するために、コードに電子署名するプロセスです。開発者は、証明機関（CA）から発行された証明書を使用するか、自己署名入り証明書を作成することによって、AIR アプリケーションに署名できます。

認定されている証明機関（CA）が発行した証明書を使用して AIR ファイルに電子署名すると、インストールしているアプリケーションが誤ってまたは悪意を持って変更されていないことをユーザーに確実に保証することができます。認定された証明機関（CA）が発行した証明書を使用して AIR ファイルに電子署名すると、開発者は署名者（発行者）と見なされます。AIR では、Verisign および Thawte 証明機関によって発行されたコード署名証明書が認識されます。開発者が Verisign または Thawte の証明書を使用して AIR ファイルに署名している場合、AIR アプリケーションインストーラーによってインストール中に発行者名が表示されます。

AIR アプリケーションインストーラーによってインストール中に発行者名が表示されるのは、信頼できる証明書、またはインストールされているコンピューターで信頼されている証明書にチェーン化されている証明書を使用して AIR アプリケーションに署名している場合です。証明機関（CA）では、信頼性の高い証明書を発行する前に、実績のあるプロセスを使用して発行者または開発者の ID を検証します。

開発者は、自分自身で作成した、自己署名入り証明書を使用して AIR アプリケーションに署名することもできます。ただし、AIR アプリケーションインストーラーでは、このようなアプリケーションは発行者が検証されていないアプリケーションとして表示されます。

AIR ファイルに署名が行われると、電子署名がインストールファイルに含まれます。署名には、署名後に AIR ファイルが変更されていないことを検証するために使用されるパッケージのダイジェストと、発行者 ID を検証するために使用する署名証明書に関する情報が含まれます。

AIR では、オペレーティングシステムの証明書ストアを通じてサポートされる公開キーインフラストラクチャ（PKI）を使用します。AIR アプリケーションがインストールされているコンピューターでは、発行側の情報を検証するために、AIR アプリケーションの署名に使用されている証明書そのものを信頼するか、信頼できる証明機関の証明書にリンクしている証明書のチェーンを信頼する必要があります。

AIR ファイルが、信頼できるルート証明書（およびこれには通常すべての自己署名入り証明書が含まれる）のいずれにもチェーン化されていない証明書で署名されている場合、発行者側の情報は検証できません。AIR は、AIR ファイルが署名後に変更されていないか確認できますが、ファイルの実際の作成者および署名者を検証することはできません。

コード署名プロセスおよび使用できる証明書の形式について詳しくは、開発者向けドキュメントを参照してください。

モバイルプラットフォームでのコード署名

モバイルプラットフォームでは、AIR アプリケーションはプラットフォームの規則や要件に従って署名されます。開発者は AIR SDK のツールやモバイルプラットフォームの要件を満たす証明書を使用してアプリケーションに署名します。モバイル AIR アプリケーションのインストールは、AIR ランタイムではなく、デバイスのオペレーティングシステムによって処理されます。そのため、AIR では、アプリケーションの署名も証明書の所有者の ID も検証されません。

関連項目

[AIR アプリケーションへの署名](#)

セキュリティサンドボックス

AIR には、AIR アプリケーションの各ファイルの権限を定義する包括的なセキュリティアーキテクチャが用意されています。これには、アプリケーションと共にインストールされるファイルと、アプリケーションによって読み込まれるその他のファイルの両方が含まれます。権限はその生成元に従ってファイルに付与され、サンドボックスと呼ばれる論理的なセキュリティグループに割り当てられます。

アプリケーションと共にインストールされるファイルはアプリケーションディレクトリと呼ばれるディレクトリに配置されます。このため、デフォルトではすべての AIR API にアクセスできるアプリケーションサンドボックスと呼ばれるセキュリティサンドボックスに配置されます。AIR API の中には、アプリケーションリソースディレクトリ以外のソースのコンテンツ（つまり、アプリケーションと共にインストールされたものではないファイル）から利用できるようにした場合に、多大なセキュリティリスクを発生させる API が含まれます。

サンドボックスの AIR セキュリティモデルは、Flash Player セキュリティモデルと追加のアプリケーションサンドボックスで構成されています。アプリケーションサンドボックスに含まれないファイルには、Flash Player セキュリティモデルで指定されているセキュリティ制限と同様のセキュリティ制限が適用されます。

ランタイムはこれらのセキュリティサンドボックスを使用して、ファイルがアクセスするデータの範囲や、ファイルが実行する処理を定義します。ローカルのセキュリティを維持するために、各サンドボックス内のファイルは他のサンドボックスのファイルと分離されています。例えば、外部のインターネット URL から AIR アプリケーションに読み込まれた SWF ファイルはリモートサンドボックスに配置されます。デフォルトでは、このファイルは、アプリケーションサンドボックスに割り当てられるアプリケーションディレクトリ内のファイルに対してスクリプトを実行するための権限を持ちません。

注意： iOS では、ダウンロードしたコードの実行は許可されません。

アプリケーションサンドボックス内のコンテンツの権限

アプリケーションがインストールされると、AIR インストーラーファイルに含まれているすべてのファイルがユーザーのコンピューターのアプリケーションディレクトリにインストールされます。アプリケーションの実行時にアプリケーションディレクトリツリー内のすべてのファイルがアプリケーションサンドボックスに割り当てられます。アプリケーションサンドボックス内のコンテンツには、ローカルファイルシステムとのやり取りを含め、AIR アプリケーションで使用できる完全な権限が与えられます。

AIR アプリケーションの多くは、ローカルにインストールされたこれらのファイルのみを使用してアプリケーションを実行します。ただし、AIR アプリケーションがアプリケーションディレクトリ内のファイルだけに制限されるわけではなく、どのソースのどの種類のファイルでも読み込むことができます。これにはユーザーのコンピューターのローカルファイルだけでなく、ローカルネットワークやインターネットなど使用可能な外部ソースのファイルも含まれます。ファイルの種類はセキュリティ制限に影響しません。読み込まれた HTML ファイルは、同じソースから読み込まれた SWF ファイルと同じセキュリティ権限を持ちます（ただし、アプリケーションサンドボックス内のコンテンツでは、サンドボックスの外部から JavaScript ファイルを読み込むことはできません。詳しくは、開発者向けドキュメントを参照してください）。

アプリケーションセキュリティサンドボックス内のコンテンツは、他のサンドボックス内のコンテンツでは使用できない AIR API にアクセスできます。例えば、アプリケーションセキュリティサンドボックス内のコンテンツのみが、ローカルファイルシステムに対して読み取りおよび書き込みを実行できます。

JavaScript の手法には、文字列を動的に実行可能コードに変換できるものがあります。アプリケーションセキュリティサンドボックス内のコンテンツは、コードがアプリケーション URL から読み込んでいるときにのみこれらの手法を使用できます。アプリケーションサンドボックス内でこれらの手法を使用すると、セキュリティリスクが発生します。例えば、ネットワークサンドボックスから読み込まれた文字列をアプリケーションで不用意に実行する可能性があります。また、その文字列には、ユーザーのコンピューターのファイルの削除または変更を行ったり、信頼できないネットワークドメインにローカルファイルのコンテンツを報告したりする悪質なコードが含まれている場合があります。詳しくは、開発者向けドキュメントを参照してください。

注意：モバイル AIR アプリケーションでは、HTML と JavaScript をアプリケーションサンドボックスに読み込むことはできません。モバイル AIR アプリケーションでは、このようなコンテンツをシステムの Web コントロールを使用して表示します。このコントロールには、デフォルトのシステムの Web ブラウザーと同じセキュリティに関する考慮事項があります。

非アプリケーションサンドボックス内のコンテンツの権限

ネットワークやインターネットから読み込んだファイルは、非アプリケーションサンドボックスに割り当てられます。このようなコンテンツは、Web ブラウザー（Flash Player）内で実行される SWF コンテンツや、Web ブラウザー内で実行される HTML コンテンツと同じ権限および制限で動作します。

アプリケーションセキュリティサンドボックスのコンテンツとは異なり、非アプリケーションセキュリティサンドボックスの HTML コードは、JavaScript のメソッドを使用して、動的に生成されたコードをいつでも実行できます。

非アプリケーションサンドボックス内のコードは、アプリケーションの機能を提供する、権限が付与された AIR API にアクセスできません。

詳しくは、開発者向けドキュメントを参照してください。

ファイルシステムへのアクセス

Web ブラウザーで実行中のアプリケーションは、ユーザーのローカルファイルシステムとのやり取りのみが制限されます。Web ブラウザーは、Web コンテンツを読み込んでもユーザーのコンピューターが改ざんされないことを保証するセキュリティポリシーを実装します。例えば、ブラウザーで Flash Player を使って実行されている SWF ファイルは、ユーザーのコンピューターに既に存在するファイルと直接やり取りできません。共有オブジェクトはユーザーの環境設定およびその他のデータを管理する目的でユーザーのコンピューターに書き込むことができますが、これはファイルシステム操作の制限になります。AIR アプリケーションはネイティブにインストールされるので、エンドユーザーとは異なるセキュリティコントラクトを備えています。このアプリケーションとエンドユーザーとのコントラクトは、ネイティブアプリケーションと同様にインストール時に作成され、アプリケーションがローカルファイルシステムに対して読み取りや書き込みを実行する機能が含まれます。

この自由度には開発者の大きな責任が伴います。アプリケーションの予想外の不安定さは、アプリケーションの機能だけでなく、ユーザーのコンピューターの完全性も危険にさらします。開発者向けドキュメントの AIR のセキュリティ情報では、ベストプラクティスについて説明しています。

ユーザーのコンピューターに対する管理者の制限がない限り、AIR アプリケーションには、ユーザーのハードドライブ上のどの場所にも書き込むことができる権限が付与されます。ただし、開発者は、ランタイムが各アプリケーションに提供する、ユーザーおよびアプリケーション固有のアプリケーション記憶領域ディレクトリを使用することをお勧めします。AIR API には、開発者がアプリケーション記憶領域ディレクトリ内のデータを読み取りおよび書き込みするための便利なメソッドが用意されています。ランタイムは、各アプリケーションおよびユーザーに固有な暗号化されたローカルデータ記憶領域も提供します。これにより、アプリケーションでは、他のアプリケーションまたはユーザーによって簡単に解読されないように暗号化された形式で、ユーザーのローカルハードディスクに格納されたデータを保存および取得できるようになります。各 AIR アプリケーションに対して個別の暗号化されたローカルストアが使用され、各 AIR アプリケーションは各ユーザーに対して個別の暗号化されたローカルストアを使用します。アプリケーションでは、Web サービスに対するログイン資格情報など、セキュリティで保護する必要がある情報を格納するために暗号化されたローカルストアを使用できます。AIR は、暗号化されたローカルストアを各ユーザーに関連付けるために、Windows では DPAPI を使用し、Macintosh では KeyChain を使用します。暗号化されたローカルストアは AES-CBC 128 ビット暗号化を使用します。

Adobe AIR 2 では、対象のファイル形式がデフォルトで登録されているアプリケーションでファイルを開くことができます。例えば、MP3 ファイルを開くデフォルトのアプリケーションで MP3 ファイルを開くことができます。AIR では、特定のファイル形式を含むファイルをアプリケーションで開けないようにしてあります。その種のファイル形式のファイルを開くと、コードが実行される可能性があります。例えば、Windows の EXE ファイルなどです。使用制限のあるファイル形式

は、『[Adobe Flash Platform 用 ActionScript 3.0 リファレンスガイド](#)』に記載されています。ただし、ネイティブインストーラーでインストールする拡張デスクトップ AIR アプリケーションではあらゆる形式のファイルを開くことができます（拡張デスクトップアプリケーションについて詳しくは、8 ページの「[ネイティブプロセスとの通信](#)」を参照してください）。

ネイティブプロセスとの通信

Adobe AIR 2 では、デスクトップ AIR アプリケーションの実行や他のネイティブプロセスとの通信を、コマンドライン経由で行うことができます。例えば、AIR アプリケーションでは、プロセスを実行し、標準の入出力ストリームを使用してそのプロセスと通信できます。

ネイティブプロセスと通信するには、開発者はネイティブインストーラーでインストールされるように AIR アプリケーションをパッケージ化します。ネイティブインストーラーのファイルタイプは、そのインストーラーが作成されたオペレーティングシステムに固有のものです。

- Mac OS では DMG ファイルです。
- Windows では EXE ファイルです。
- Linux では RPM または DEB パッケージです。

これらのアプリケーションは、拡張デスクトッププロファイルアプリケーションとして知られています。

これらのアプリケーションをパッケージ化する際、開発者はコード署名証明書を使用してアプリケーションに署名します。同様の証明書を、標準のデスクトップ AIR アプリケーションの署名に使用する証明書として使用します。

ネイティブプロセス API は、ユーザーのシステム上で任意の実行可能ファイルを実行できます。開発者は AIR ドキュメントを参照すると、ネイティブプロセス API の使用に関する情報が得られます。開発者はコマンドの構成および実行の際は注意が必要です。アプリケーションがネイティブプロセスに送信されるデータを検証する必要があります。

Windows 上では、AIR は拡張デスクトップアプリケーションが直接 .bat ファイルを実行できないようにします。.bat ファイルへのコマンドライン引数には悪意のある余分な文字が注入されている可能性があります。このような注入は、有害なまたは安全でないアプリケーションを実行する cmd.exe アプリケーション (.bat ファイルを実行する) をもたらす可能性があります。

信頼されないコンテンツの安全な使用

アプリケーションサンドボックスに割り当てられていないコンテンツは、ランタイムのセキュリティ基準を満たす場合のみ AIR アプリケーションに追加のスクリプト機能を提供できます。ここでは、非アプリケーションコンテンツでの AIR セキュリティコントラクトについて説明します。

AIR アプリケーションは、Flash Player のブラウザープラグインが信頼されないコンテンツのスクリプトによるアクセスを制限するよりも厳格に、非アプリケーションコンテンツのスクリプトによるアクセスを制限します。例えば、ブラウザー内の Flash Player では、SWF ファイルで System.allowDomain() メソッドを呼び出して、指定したドメインから読み込まれた SWF コンテンツへのスクリプトによるアクセスを許可することができます。このメソッドの呼び出しは、ユーザーのファイルシステム内の非アプリケーションファイルへの不適切なアクセスを許可することになるので、アプリケーションセキュリティサンドボックス内のコンテンツについては許可されていません。

アプリケーションコンテンツと非アプリケーションコンテンツとの間をスクリプト化する AIR アプリケーションではさらに複雑なセキュリティの調整が行われます。アプリケーションサンドボックス内に含まれていないファイルは、サンドボックスブリッジを使用したアプリケーションサンドボックス内のファイルのプロパティとメソッドへのアクセスのみが許可されます。サンドボックスブリッジはアプリケーションコンテンツと非アプリケーションコンテンツ間のゲートウェイとして機

能し、2つのファイル間の明示的な操作を可能にします。サンドボックスブリッジを正しく使用すると、セキュリティの追加のレイヤーが提供され、非アプリケーションコンテンツが、アプリケーションコンテンツの一部となるオブジェクト参照にアクセスするのを制限します。

サンドボックスブリッジのメリットについて、例を使って説明します。AIR ミュージックストアアプリケーションは、独自の SWF ファイルを作成する必要がある広告主に API を提供し、ストアアプリケーションはこのファイルを使用して通信できます。ストアは、ストアが提供するアーティストと CD を探すためのメソッドを広告主に提供する必要がありますが、セキュリティ上の理由でサードパーティの SWF ファイルからメソッドとプロパティを分離する必要があります。

サンドボックスブリッジはこの機能を提供できます。デフォルトでは、ランタイムで外部から AIR アプリケーションに読み込まれたコンテンツには、メインアプリケーションのどのメソッドまたはプロパティへのアクセス権もありません。カスタムのサンドボックスブリッジ実装を使用すると、開発者はこれらのメソッドやプロパティを公開せずにリモートコンテンツにサービスを提供できます。サンドボックスブリッジは信頼されるコンテンツと信頼されないコンテンツ間の制限された通路となります。

サンドボックスブリッジの使用に関する完全な詳細については、次を参照してください。

- ActionScript (Flash および Flex) 開発者は、『ActionScript 3.0 開発ガイド』の [AIR のセキュリティ](#) を参照してください。
- Ajax 開発者は、『Adobe AIR 用 HTML 開発ガイド』の [AIR のセキュリティ](#) を参照してください。

Android デバイスのセキュリティ

すべてのコンピューティングデバイスと同様、Android の AIR は、ネイティブセキュリティモデルに準拠しています。同時に、AIR では、安全なインターネット接続アプリケーションを開発者が簡単に作成できるようにするための独自のセキュリティ規則を維持しています。

Android 上の AIR アプリケーションでは Android パッケージ形式を使用するので、インストールには Android セキュリティモデルが適用されます。AIR アプリケーションインストーラーは使用されません。

Android セキュリティモデルには、次に示す主要な 3 つの要素があります。

- 権限
- アプリケーションの署名
- アプリケーションユーザー ID

Android 権限

Android の多くの機能は、オペレーティングシステムの権限メカニズムによって保護されます。保護された機能を使用するには、アプリケーションに権限が必要であることを AIR アプリケーション記述子で宣言する必要があります。ユーザーがアプリケーションをインストールしようとする、Android オペレーティングシステムでは、インストールを続行するために必要な権限がすべて表示されます。

ほとんどの AIR アプリケーションでは、アプリケーション記述子に Android 権限を指定する必要があります。デフォルトでは、権限は含まれていません。AIR ランタイムを通じて公開される Android の保護された機能を使用するには、次の権限が必要です。

ACCESS_COARSE_LOCATION アプリケーションが、Geolocation クラスを介して Wi-Fi およびセルラーネットワーク位置データにアクセスすることを許可します。

ACCESS_FINE_LOCATION アプリケーションが Geolocation クラスを介して GPS データにアクセスすることを許可します。

ACCESS_NETWORK_STATE and ACCESS_WIFI_STATE アプリケーションが NetworkInfo クラスを介してネットワーク情報にアクセスすることを許可します。

CAMERA アプリケーションがカメラにアクセスすることを許可します。

INTERNET アプリケーションがネットワーク要求を行うことを許可します。また、リモートデバッグも許可します。

READ_PHONE_STATE 着信呼び出しがあったときに、AIR ランタイムがオーディオをミュートすることを許可します。

RECORD_AUDIO アプリケーションがマイクにアクセスすることを許可します。

WAKE_LOCK および DISABLE_KEYGUARD アプリケーションが、SystemIdleMode クラス設定を使用してデバイスのスリープ状態を防ぐことを許可します。

WRITE_EXTERNAL_STORAGE アプリケーションがデバイスの外部メモリカードへ書き込むことを許可します。

アプリケーションの署名

Android プラットフォーム用に作成されたすべてのアプリケーションパッケージには署名が必要です。Android 上の AIR アプリケーションはネイティブの Android APK 形式でパッケージ化されるので、アプリケーションの署名は AIR の規則ではなく Android の規則に従って行われます。Android と AIR で使用するコード署名はよく似ていますが、次のような大きな違いがあります。

- Android では、開発者が秘密キーを所有しているかどうか署名によって検証されますが、開発者の ID の検証には署名が使用されません。
- Android マーケットに送信されるアプリケーションの場合、少なくとも 25 年間有効な証明書が必要です。
- Android では、別の証明書へのパッケージの署名の移行がサポートされません。更新の署名が別の証明書によって行われている場合、ユーザーは、更新されたアプリケーションをインストールする前に元のアプリケーションをアンインストールする必要があります。
- 同じ証明書を使用して署名された 2 つのアプリケーションでは、互いにキャッシュやデータファイルにアクセスできる共有 ID を指定できます（このような共有は AIR では容易ではありません）。

アプリケーションユーザー ID

Android では Linux カーネルを使用します。インストールされたすべてのアプリケーションには Linux タイプのユーザー ID が割り当てられます。この ID によって、ファイルアクセスなどの操作の権限が決定されます。アプリケーションディレクトリ、アプリケーション記憶領域ディレクトリ、および一時ディレクトリ内のファイルは、ファイルシステムの権限によってアクセスから保護されます。外部ストレージへ書き込まれるファイル（つまり、SD カード）は、SD カードがマスタストレージデバイスとしてコンピューターにマウントされている場合、他のアプリケーションまたはユーザーが読み込み、変更、削除を行うことができます。

インターネット要求とともに受信されるクッキーは、AIR アプリケーション間で共有されません。

関連項目

[Android : Security and Permissions](#)

アプリケーションのインストール

デフォルトで、Android 上の AIR アプリケーションは、アドビが管理および更新する共有ランタイムライブラリを使用します。AIR 3 からは、「キャプティブ」ランタイムでアプリケーションをバンドルできます。デバイス上には共有 AIR ランタイムも存在するかもしれませんが、キャプティブランタイムでインストールされたアプリケーションは、共有 AIR ランタイムではなくこのバージョンのランタイムを使用します。新しいバージョンの AIR ランタイムの公開時に、キャプティブランタイムは自動的に更新されません。

重要：キャプティブランタイムを使用する場合は、関連するセキュリティのアップデートが公開された際に、自身でランタイムを更新する必要があります。

Android における暗号化されたデータ

Android 上の AIR アプリケーションでは、ビルトイン SQL データベースに用意されている暗号化オプションを使用して、暗号化されたデータを保存できます。

EncryptedLocalStore クラスは、データの保存に使用できますが、そのデータは暗号化されません。代わりに、Android セキュリティモデルはアプリケーションユーザー ID を使用して他のアプリケーションからのデータを保護します。共有のユーザー ID を使用するアプリケーションと、同じコード署名証明書で署名されたアプリケーションは、同じ暗号化されたローカルストアを使用します。

重要：ルート権限のある電話では、ルート権限を使用して実行するアプリケーションからは他のどのアプリケーションのファイルにもアクセスできます。したがって、暗号化されたローカルストアを使用して保存されるデータは、ルート権限のあるデバイス上では保護されません。

iOS デバイスのセキュリティ

iOS では、AIR はネイティブセキュリティモデルに準拠します。同時に、AIR では、安全なインターネット接続アプリケーションを開発者が簡単に作成できるようにするための独自のセキュリティ規則を維持しています。

iOS 上の AIR アプリケーションでは iOS パッケージ形式を使用するので、インストールには iOS セキュリティモデルが適用されます。AIR アプリケーションインストーラーは使用されません。さらに、iOS デバイスでは個別の AIR ランタイムが使用されません。各 AIR アプリケーションには、機能に必要なすべてのコードが含まれています。

重要：ランタイムコードはアプリケーションの一部として含まれているので、AIR の新しいバージョンのリリース時に、自動的に更新されません。関連するセキュリティのアップデートが公開されたら、自身でアプリケーションを更新する必要があります。

アプリケーションの署名

iOS プラットフォーム用に作成されたすべてのアプリケーションパッケージには署名が必要です。iOS 上の AIR アプリケーションはネイティブの iOS IPA 形式でパッケージ化されるので、アプリケーションの署名は AIR の要件ではなく iOS の要件に従って行われます。iOS と AIR で使用するコード署名はよく似ていますが、次のような大きな違いがあります。

- iOS では、Apple が発行した証明書をアプリケーションの署名に使用する必要があります。他の証明機関の証明書は使用できません。
- iOS の場合、Apple が発行した配布証明書は、通常 1 年間有効です。

HTML セキュリティ

HTML コンテンツでは、主に JavaScript による動的に生成されるコードの作成機能のために、SWF ベースのコンテンツとは異なるセキュリティの考慮事項があります。eval() 関数を呼び出すときに作成されるような、動的に生成されたコードがアプリケーションサンドボックス内で許可されると、セキュリティリスクが発生することがあります。例えば、ネットワークサンドボックスから読み込まれた文字列をアプリケーションで不用意に実行する可能性があります。また、その文字列には、ユーザーのコンピューターのファイルの削除または変更を行ったり、信頼できないネットワークドメインにローカルファイルのコンテンツを報告したりする悪質なコードが含まれている場合があります。

動的なコードを生成するには、次のような方法があります。

- eval() 関数を呼び出します。
- innerHTML プロパティを設定するか、または DOM 関数を呼び出してスクリプトタグを挿入してリソース外のスクリプトを直接読み込みます。

- innerHTML プロパティを設定するか、または DOM 関数を呼び出して、(src を使用してスクリプトを読み込むのではなく) インラインコードを含む script タグを挿入します。
- アプリケーションサンドボックス内のコンテンツについて、script タグの src を、アプリケーションリソースディレクトリ以外にあるファイルに設定します。
- javascript URL スキームを使用します (href="javascript:alert("Test")" など)。

アプリケーションセキュリティサンドボックス内のコードは、アプリケーションディレクトリからコンテンツを読み込んでいるときにのみこれらのメソッドを使用できます。これによって、すべての AIR API にアクセスできるアプリケーションサンドボックス内のコードが、潜在的に信頼されないソースのスクリプトを実行することを防止できます。

非アプリケーションセキュリティサンドボックスのコンテンツは、これらのメソッドを使用して動的コードを生成できます。ただし、直接 AIR API にアクセスできるわけではありません。AIR のサンドボックスブリッジ機能によって、非アプリケーションセキュリティサンドボックス内のコードが、アプリケーションコードによって制限および決定された方法で、アプリケーションサンドボックス内のコードとやり取りするための手段が提供されます。

AIR アプリケーションは、(ファイルやネットワークのソースから読み込むのではなく) HTML スtring変数から HTML コンテンツを生成できます。ただしデフォルトでは、Stringによって生成された HTML コンテンツには、アプリケーションサンドボックスの権限が付与されません。これにより、潜在的に安全でないインターネットのソースから取得される Stringコンテンツに、アプリケーションが不用意にアクセスしないようにします。

注意: モバイルデバイスの AIR は、ホストオペレーティングシステムが提供する Web コントロールを使用します。このコントロールで実行するコンテンツは AIR API にアクセスできないので、アプリケーションセキュリティサンドボックスに読み込んだり、アプリケーションセキュリティサンドボックスで実行したりできません。

HTML セキュリティについて詳しくは、開発者向けドキュメントの「AIR のセキュリティ」を参照してください。

- ActionScript (Flash および Flex) 開発者は、『ActionScript 3.0 開発ガイド』の [AIR のセキュリティ](#) を参照してください。
- Ajax 開発者は、『Adobe AIR 用 HTML 開発ガイド』の [AIR のセキュリティ](#) を参照してください。
また、[Adobe AIR HTML のセキュリティ](#) のホワイトペーパーも参照してください。

その他のセキュリティに関する考慮事項

AIR アプリケーションが Web テクノロジーを使用して構築されていても、開発者はそれらの AIR アプリケーションがブラウザのセキュリティモデル内で機能していないことに気づくことが重要です。つまり、故意にまたは誤ってローカルシステムに損害を与える可能性がある AIR アプリケーションを構築する場合があります。AIR はリスクを最小化しようと試みますが、脆弱性が取り込まれる可能性はまだあります。ここでは、重要なセキュリティ上の潜在的なリスクについて説明します。開発者向けのドキュメントでは、このようなリスクを回避するアプリケーションを構築するためのベストプラクティスを示しています。

ファイルをアプリケーションセキュリティサンドボックスにインポートした場合のリスク

アプリケーションサンドボックスがランタイムのすべての権限を持つコンテンツ。開発者は次の点に注意する必要があります。

- ファイルは必要な場合にのみ AIR ファイル (インストール済みアプリケーション内) に含めます。
- スクリプトファイルは、動作が完全に把握され、信頼される場合にのみ AIR ファイル (インストール済みアプリケーション内) に含めます。

- ネットワークソースのデータを AIR API のメソッドのパラメーターとして使用しないでください。これによりコードが実行される可能性があります。

Adobe AIR では、アプリケーションサンドボックス内のコンテンツがネットワークソースのデータをコードとして使用することによって、悪意のあるコードが誤って実行されることを防ぎます。これには、ActionScript Loader.loadBytes() メソッドや JavaScript eval() 関数の使用が含まれます。

外部ソースを使用してパスを判断する場合のリスク

AIR アプリケーションは、外部のデータまたはコンテンツを使用すると、改ざんされる可能性があります。このため、アプリケーションでネットワークまたはファイルシステムのデータを使用する場合は特に注意する必要があります。信頼の責任は最終的には開発者と、開発者が作成するネットワーク接続にあります。外部データの読み込みは本来リスクを伴うものであるため、機密データの操作への入力には使用しないでください。開発者は次の操作を行わないことをお勧めします。

- ネットワークソースのデータを使用したファイル名の決定
- ネットワークソースのデータを使用した、個人情報の送信または他のアプリケーションの起動にアプリケーションで使用される URL の作成

保護されていない資格情報を使用、保存または送信する場合のリスク

ユーザー資格情報をユーザーのローカルファイルシステムに保存することは本来リスクを伴い、これらの資格情報が改ざんされる可能性があります。開発者は次の点に注意する必要があります。

- 資格情報をローカルに保存する必要がある場合は、ローカルファイルシステムに書き込むときに資格情報を暗号化します。Adobe AIR は、インストール済みの各アプリケーションに固有の暗号化された記憶領域を提供します。詳しくは、開発者向けのドキュメントを参照してください。
- ネットワークソースが信頼でき、送信プロトコルが TLS (Transport Layer Security)、https: または SecureSocket クラスを使用している場合以外は、暗号化されていないユーザー資格情報をネットワークソースに送信しないようにしてください。
- 資格情報の作成時にデフォルトのパスワードを指定しないでください。パスワードはユーザー自身が作成できるようにしてください。デフォルトを使用するユーザーはその資格情報を、デフォルトパスワードを知っている攻撃者に公開することになります。

リモートデータの送受信でのリスク

インターネットを介しての情報をやり取りでは、情報漏洩や改ざんの被害を受ける可能性があります。TLS (Transport Layer Security) プロトコルまたはこれより古い SSL (Secure Socket Layer) プロトコルを使用して、サーバーとクライアントシステムとの間の通信を暗号化できます。HTTP の通信を保護するには、HTTPS プロトコルを使用します。TCP ソケットの通信を保護するには、SecureSocket クラスを使用します。どちらの場合も、AIR アプリケーションがデータの送受信を安全に行うために、データを提供するサーバーでは TLS または SSL を使用する必要があります。AIR 2 では SecureSocket クラスが追加されました。このクラスは AIR アプリケーションサンドボックスで実行するコンテンツに使用できます。開発者は次の点に注意する必要があります。

- サーバーでホストされている機密データへのアクセス権の授与は、TLS プロトコルの最新版を使用して行います (現時点では、AIR は TLS バージョン 1 および SSL バージョン 4 をサポートします)。
- HTTP プロトコルを使用する機密データを送受信する場合は、HTTPS を使用します。
- TCP ソケットを使用する機密データを送受信する場合は、SecureSocket クラスを使用します。

ダウングレード攻撃のリスク

アプリケーションのインストール処理中に、ランタイムはアプリケーションのバージョンが現在インストールされていないことを確認します。アプリケーションが既にインストールされている場合、ランタイムは既存のアプリケーションのバージョンストリングを、インストールしようとしているバージョンと比較します。このストリングが異なる場合、ユーザーはインストールのアップグレードを選択できます。ランタイムは、新しくインストールされるバージョンが古いバージョンよりも新しいことを保証するわけではなく、それが異なるバージョンであるということだけを保証します。攻撃者は古いバージョンをユーザーに配布して、セキュリティの弱点をくぐり抜ける可能性があります。このリスクは軽減することができ、リスクを回避するためのバージョンスキームおよびアップデート確認の実装に関するベストプラクティスについては、開発者向けのドキュメントを参照してください。