

アプリのライフサイクル を理解する：アップデート と導入の自動化



アプリの調達、アップデート、そしてユーザに配慮したプロセスの構築など、アプリのライフサイクル全体を通じた管理はIT部門の重要な仕事です。とはいえ、その重要性が見逃されていたり、現状に満足したまま放置されることも多いものです。しかし実際は、ワークフローに問題があることがあまりにも多く、ITの仕事を増やしたり頭痛の種になったりするほか、長期的にユーザエクスペリエンスを阻害する可能性もあります。

アプリのライフサイクル管理のワークフローにおける各ステップへのアプローチを見直すことで、古いソフトウェアに起因する互換性の問題やセキュリティリスクに目を向けながら、さらに優れたユーザエクスペリエンスを提供することができます。そして、互換性の問題にタイムリーに対応する上でもっとも重要な要素のひとつが、パッチ管理です。

本ホワイトペーパーのトピック

- アプリの導入とアップデートに関する基礎知識
- ソフトウェアにパッチを適用すべき理由
- Jamf ProのAppインストーラ
- Jamf Proを使ったパッチ管理

アプリの導入とアップデートに関する基礎知識

アプリケーションのライフサイクルの大部分を占めるのが、アップデートと導入の作業です。Apple Business ManagerやApple School Managerで一括購入されたアプリは、新しいバージョンがリリースされた段階で自動的にアップデートされますが、業務用アプリケーションの中にはMac App Store以外の場所（Google ChromeやAdobe Suiteなど）から入手されたものも多くあります。これらのアプリはアップデートや導入に少し手間がかかります。

こういったアプリをアップデートするためにはパッチ管理が必要になりますが、IT管理者が手作業で行うため時間がかかります。まず、ソフトウェア環境においてなぜパッチ管理が必要なのかについて見ていきましょう。

ソフトウェアにパッチを適用すべき理由

ユーザが必要とする新機能が含まれたソフトウェアアップデートの利用

ソフトウェアのメジャー（またはマイナー）アップデートには新機能が含まれていることが多く、ITが認識する前にユーザからリクエストされることも少なくありません。

バグ修正が含まれたバージョンにアップデートし、サポートデスクへの問い合わせ件数を削減

業務に必要なアプリに新たな機能が追加されることはユーザにとっても嬉しいことですが、バグが減ることでサポートデスクへの問い合わせ件数が減ればITにとっても大きなメリットとなります。

セキュリティの脆弱性への迅速な対応

ServiceNowの調査によると、セキュリティの脆弱性を解決するためのパッチがリリースされたあと再度サイバー攻撃が試みられるまでに平均で43日かかり、データ侵害を経験した回答者の60%がパッチを適用していれば侵害を回避できた可能性があると答えています。

パッチがタイムリーに適用されないケース

前述のServiceNowの調査によると、パッチの適用が遅れる理由として以下の点が多く挙げられます。

- パッチ適用に必要なリソースの欠如または不足
- インストールされているアプリやデバイスを一元的に把握できていない
- パッチ適用のためのダウンタイムを手配できない

幸いにも、Jamf Proがあれば上記の3つの点を解決することが可能です。





Jamf Proでアプリをアップデートする方法

Jamf Proでは、パッチ管理を成功させるための2つの異なるワークフローが用意されています。1つ目はAppインストーラです。これはパッチ管理プロセスを自動化することで、新しいバージョンの調達と導入にかかる時間を短縮するものです。2つ目のオプションであるポリシーを用いたワークフローは、Appインストーラの場合に比べて手作業が必要になりますが、それが唯一の選択肢だった頃からJamfを利用しているApple管理者にとっては馴染み深いものかもしれません。



バージョン履歴

バージョン履歴とは、ソフトウェアのバージョンとリリース日の情報をまとめたものです。バージョン履歴は、ソフトウェアに適用すべきパッチがリリースされているかどうかを確認するには便利ですが、該当するソフトウェアにパッチを適用するか否かを管理者が判断する上で必要な情報が十分ではない場合があります。

Jamfでパッチ管理を行う場合、パッチ適用に不可欠な詳細情報がソフトウェアのバージョンごとに含まれており、例えば適用が可能なmacOSバージョンやmacOSデバイスの再起動の必要性、適用に際して終了しなければならないアプリケーションなどについて知ることができます。

Appインストーラによるパッチ適用ワークフローの自動化

アプリを最新の状態に保つ方法のひとつに、アップデートの自動化があります。Jamfのアプリカタログに含まれるAppインストーラは、サードパーティ製のMac用アプリの導入とアップデートを効率的に行う方法を提供し、アプリケーションタイトルの調達、導入、メンテナンスといった面倒な作業の省略を実現します。

Appインストーラでアプリのアップデートを自動化する

App Storeのアプリとサードパーティーのアプリを一元的に管理

App Storeは、ユーザが必要とするアプリケーションが一箇所に集まっているという点で、非常に画期的なマーケットプレイスと言えます。一方、Jamf Proには、Mac App Storeとサードパーティーの両方のアプリケーションを簡単に調達できる独自のアプリカタログが用意されています。また、Appインストーラのシンプルな「自動アップデート」のワークフローは、アプリのアップデートの負担を軽減してくれます。

管理者は、パッチポリシーの対象にしたいアプリをJamfのアプリカタログから選択できるほか、外部のパッチ定義フィードも利用できます。また、カタログに含まれる多数のアプリに加え、[Jamf Title Editor](#)を利用すれば、自社アプリやカスタムアプリを作成してJamfのアプリカタログ内で管理したり、アプリのタイトル定義に含まれる情報を変更したりすることも可能です。

STEP 1

アプリの調達、ホスティング、パッケージ化の自動化

Jamfがアプリのアップデートの有無を監視し、最新アップデートがあればベンダーから直接ダウンロードし、必要に応じて再パッケージ化し、配布先にアップロードしてくれます。アップロードが完了したら、Jamf Pro経由でそのパッケージがパッチ定義に紐づけられます。

STEP 2

パッケージを通じたアプリのセキュリティ担保

アプリを安全に導入するには、まずパッチ定義の整合性が検証されなければなりません。整合性が確認されたら、アプリのアップデートは管理者によって指定されたスマートグループ内の該当するMacに自動的に導入されます。この一連の動作はバックグラウンドで行われるため、ユーザ側のアクションは一切必要ありません。



Jamf Proを使ったパッチ管理

その他にも、Jamf Proのパッチポリシーを使用してアプリにパッチを適用したり、この方法をAppインストーラと組み合わせたりすることもできます。この場合、管理者はパッチポリシーの対象にしたいアプリをJamfのアプリカタログから選択します。

Jamf Proでは、登録されたデバイスに関する豊富な情報をインベントリとして収集することが可能です。そして、そこにはインストール済みのソフトウェアタイトルとそのバージョン情報も含まれています。

バージョン履歴は、アプリが古くなったことを把握する上で、管理者にとって重要な情報となります。Jamf Proのパッチ管理機能を活用することで、管理者はソフトウェアのバージョンデータを継続的に受信することができ、タイムリーなレポート情報だけでなく、デバイスへのアクションやパッチ適用を行うことができます。

さらに、インベントリデータとソフトウェアのバージョンデータを組み合わせて、ソフトウェアのコンプライアンスに関するレポートをタイムリーに作成し、管理者に必要な情報を提供することで、パッチの適用プロセスにつなげることができます。

バージョン情報に含まれるもの

パッチ管理を行うにあたって、Jamfはアプリケーション名(.app)とバンドル識別子(CFBundleIdentifier)を使用してソフトウェアタイトルを識別し、アプリケーションのショートバージョン文字列(CFBundleShortVersionString)を使用してソフトウェアタイトルのバージョンを特定します。

例: Google Chromeの場合

App name: Google Chrome.app

CFBundleIdentifier:
com.google.Chrome

CFBundleShortVersionString:
88.0.4324.150

ソフトウェアによっては、非標準的な場所にある構成ファイルなど、バージョン情報を報告するために別の方法を使用していたり、コマンドラインからしかアクセスできなかったりする場合があります。このような場合、Jamf Proは拡張属性を使用し、インベントリ情報を追加で収集してソフトウェアタイトルのバージョンをクエリします。

STEP 1

Jamf Proを使って ソフトウェアタイト ルの状態を把握

パッチ管理の対象となるソフトウェアタイトルは、目的に応じてレポートおよびアップデートの構成を行うことが可能です。ソフトウェアタイトルがバージョン情報の収集に拡張属性を必要とする場合は、データ収集プロセスの完全な透明性が保たれるよう、使用するスクリプトは承認を受けることになります。

STEP 2

レポート 作成

次にレポートを作成し、そのソフトウェアタイトルがインストールされているコンピュータを特定し、パッチが必要なソフトウェアタイトルを確認します。レポートは、ソフトウェアコンプライアンスの概要を一目で確認できるようJamf Proのダッシュボードに表示したり、.csv形式または.txt形式のファイルに出力したりできます。

お使いの環境の全体的な状態をもっとはっきりと把握するには、コンピュータの詳細検索やスマートグループを活用することをお勧めします。

STEP 3

通知

業務に不可欠なソフトウェアタイトルのアップデートが利用可能になった時に、それを管理者に通知するようJamf Proを構成することができます。また、ユーザが各自アップデートを行えるよう、ソフトウェア単位で通知することもできます。Jamf Proアカウントがプロビジョニングされている場合、セキュリティチームにも通知を送り、管理者と並行して作業を行えるようにすることが可能です。

ServiceNowが行った調査によると、62%の回答者が脆弱性の存在にまったく気づかなかったと答えており、このような対策を取っておくことは不可欠と言えます。アップデートが利用できることが通知されていれば、すぐにパッチを適用して問題を解決できたかもしれないからです。

STEP 4

パッチの 導入

Jamf Proに追加されたソフトウェアタイトルへのパッチの導入は、パッチポリシーを通じて完全に管理することができます。

これはパッチに特化したJamf Proのポリシーで、パッチを自動的に配布するかどうかを選んだり、ユーザが各自Self Serviceから入手する方法を選んだりできるほか、パッチが利用可能になったことを通知センター経由でユーザに通知するよう設定することも可能です。さらに、パッチを拒否するオプションや猶予期間を設けるオプションなども用意されており、ITの都合ではなくユーザにとって最適なタイミングでアップデートをインストールすることができます。



Jamfを利用したシンプルなアプリ管理

組織の規模が大きくなると、アプリの管理を簡素化する必要が出てきます。健全なアプリ管理の環境を維持し、アプリを最新の状態に保つことは、IT部門の重要な役割のひとつです。パッチ管理ソリューションを使ってソフトウェアタイトルを手動でアップデートする場合でも、Appインストーラ経由でプロセスを自動化する場合でも、Jamf Proがあればアプリの調達、導入、アップデートはこれまでにないほどシンプルになります。

そして、これは始まりにすぎません。Jamfは今後も、アプリのライフサイクル管理をさまざまなソフトウェア環境下で成功させるために尽力するIT管理者のために、新たなワークフローへの投資を続けていきます。

Jamfを使ったアプリのライフサイクル管理を試してみたい方は、ぜひ **無料トライアル**にお申し込みください。

または、お近くの販売代理店までお問い合わせください。