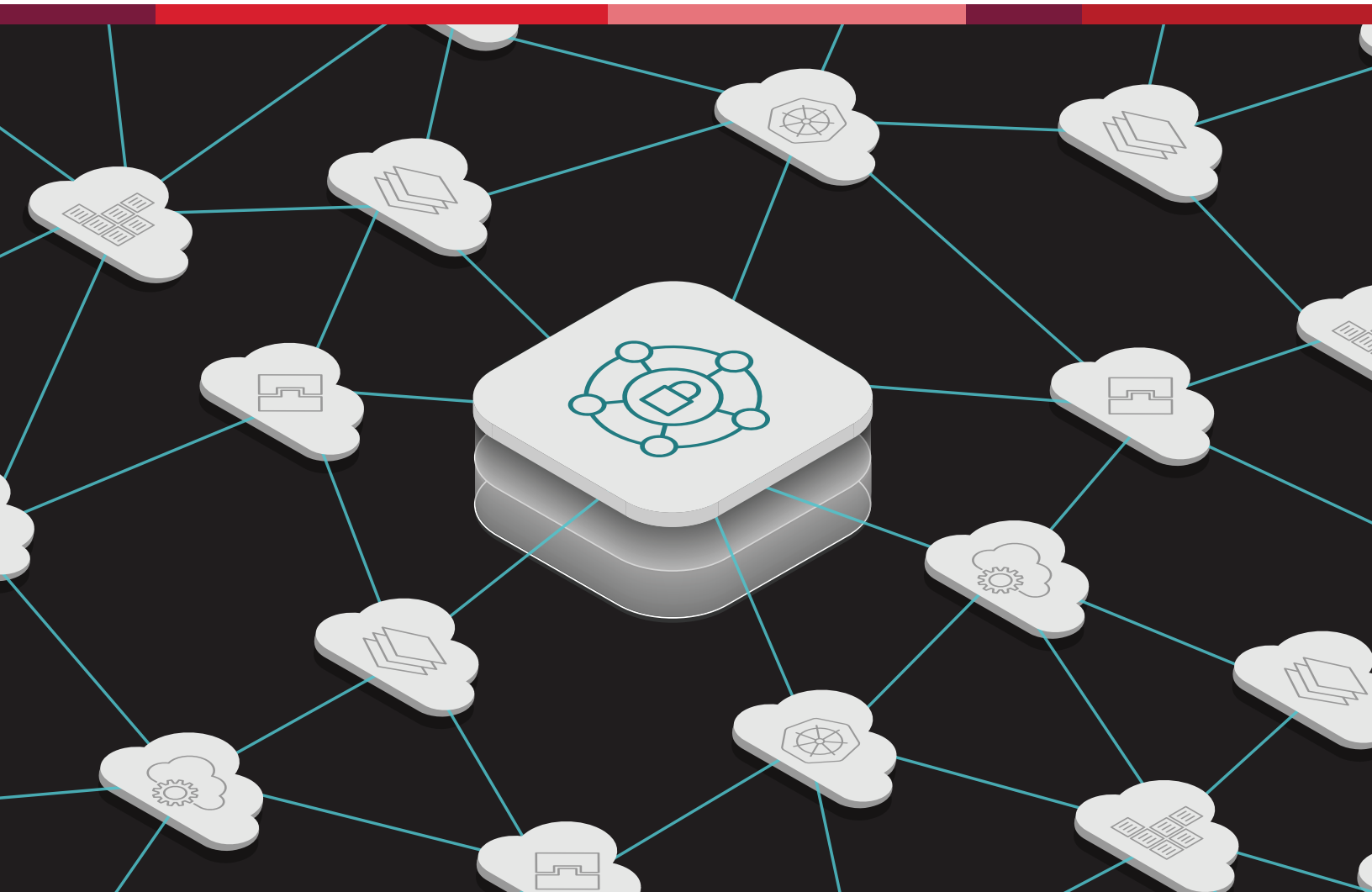
ホワイトペーパー

F5 Distributed Cloud AIP Agent の詳細

このホワイトペーパーでは、当社の Agent のコンポーネントとデータ収集、およびそれらがどのように
アプリケーションインフラストラクチャを保護するかについて説明します。



F5® Distributed Cloud App Infrastructure Protection（AIP、旧製品名：Threat Stack）は、いくつかのコンポーネントで構成されています。その中で、**F5 Distributed Cloud AIP Agent** は、ホストオペレーティングシステムとの直接的なインターフェイスとして機能します。

Agent を実稼働環境に導入する前、あるいは開発環境での試用を検討する前でも、ソフトウェアのこの部分が何をしているのか、どのように機能しているかという疑問が生じるのは当然です。このホワイトペーパーでは、Agent のコンポーネント、サポート対象システム、収集されるデータの種類、および Agent 適用に関するいくつかの簡単なユースケースを説明することで、これらの疑問を解決していきます。

概要

Agent は、Go と C で実装され、Linux または Windows Server システムの低レベルのオペレーティングシステムの活動、ファイル整合性監視（FIM）、およびユーザースペースイベントに関するデータを収集します。ネットワークトラフィックのスニッフィングやパケットインスペクションは行いません。しかし、Agent のホスト OS 中心のビューでは、ネットワーク接続を開く前に特定のユーザーが実行したコマンドや、ポート、IP アドレス、実行時に渡された関連引数を観測した、ネットワーク接続に関連するプロセスレベルのメタデータに関する貴重なコンテキストが提供されます。

Linux Roots

F5 Distributed Cloud AIP は本来、既存の Linux サブシステムを利用し、複雑なログ収集や異常行動の検出を軽減するように設計されていました。Agent 自体はいくつかのデータ収集コンポーネントで構成されていますが、Linux 監査デーモン auditd に代わる tsauditd を核としています。

Linux 監査システムに不慣れな方のために説明すると、Linux 監査システムは、ユーザースペースプロセスのセットと「カーネル側システムコール処理」用コンポーネントの 2 つの主要コンポーネントで構成されています。1 これらのユーザースペースの 1 つが auditd です。tsauditd は、ユーザースペースから既存の Linux 監査システムに接続し、Linux セキュリティモジュール（LSM）を追加でロードすることなく、カーネルシステムコールのログを記録します。実際、Agent for Linux のすべての部分はユーザースペースで実行されます。

The Agent for Linux

F5 Distributed Cloud AIP Agent for Linux は、4 つのプロセスで構成されています。これらの一部は、環境によっては無効化できますが、通常はこれらは組み合わせて実行されます。

- `tsagentd`：systemd によって生成されるワーカープロセスで、各種センサーからの入力を管理します。
- `tsauditd`：高性能かつ低遅延、最小限のコンピュートリソース使用で RAW カーネル監査イベントを消費、処理および変換する auditd の代替プロセスと、組み込みファイルシステム API（inotify と fanotify）を介してターゲット FIM を実行するプロセスです。
- `tscontainerd`：Docker コンテナからイベントを収集するためのセンサーです。

- `tskubes`: Kubernetes Events API からオーケストレーションイベントやその他のセキュリティ関連メトリクスのデータをフィルタリングするセンサーです。

導入オプション
2014 年から皆様の実稼働環境で使用されている Agent には、インフラストラクチャと運用のワークフローに合わせて組み合わせることができる、いくつかの導入オプションがあります。

- Agent を Linux ホストに `apt` または `yum` で手動インストールする。
- マシンイメージを作成し、スタートアップスクリプトで導入キーを渡す。
- Chef、Puppet、Salt または Ansible を使用してインストールを自動化する。
- Agent を独自の Docker コンテナとして導入し、オプションで Kubernetes DaemonSet に含める。このバージョンの Agent は独自のコンテナとして実行されますが、ワーカーノードごとに 1 回実行され、ホストベースの F5 Distributed Cloud AIP Agent と同じように詳細な可視化と機能が維持されます。

新しい eBPF 機能

F5 Distributed Cloud AIP Agent for Linux に、ネットワークと DNS の追加テレメトリーを報告する新しい eBPF コンポーネントが追加されました。eBPF は、カーネルの既存のコードやモジュールを変更することなく Linux カーネルの機能を安全かつ効率的に拡張することで、ランタイムでのオペレーティングシステム（OS）への機能追加を可能にします。eBPF により、イベント駆動型のカスタムコードをアプリケーションやカーネルの変更することなく OS カーネルでネイティブに実行し、ランタイムのセキュリティと可観測性を実現できます。

Agent for Windows Server

Agent for Windows Server は、ネイティブな Windows サブシステムから効率的にセキュリティシグナルを収集し、ファイル整合性監視のための独自のミニフィルタドライバを実行します。Windows Agent のコア機能は Go で書かれていますが、FIM ミニフィルタドライバは C++ で実装されます。FIM ドライバは Microsoft により署名されます。

Windows Agent は、Windows イベントログからセキュリティイベントを表示します。Sysmon がサーバー上で動作している場合、Sysmon イベントを収集して、セキュリティフォレンジックに役立つ「プロセスの作成、ネットワークへの接続、ファイル作成時刻の変更」に関するメタデータを追加するように設定できます。

導入
F5 Distributed Cloud AIP は、Windows Agent をインストールするための MSI ファイルを提供します。Windows Server GUI では、一回限りのインストールがサポートされていますが、Agent は「サイレントモード」での無人コマンドプロンプトインストールもサポートしています。

クラウドセキュリティの脅威の多くは、ホスト OS 上で存在感を確立し、それを使用する人やプロセスの行動を観測することによってのみ可視化できます。

Agent パフォーマンスに関する考慮事項

F5 Distributed Cloud AIP は、常に Agent を最適化してオーバーヘッドを最小化しようとしています。しかし、何を使い、どのように調整するかによって、監視対象の環境の CPU とメモリーへの影響は異なります。

特定の Linux ワークロードは、他のワークロードよりも多くのカーネル監査メッセージを生成します。たとえば、高い割合の子プロセスのフォークや実行などです。Agent はより大量の execve システムコールに対応しようとしますが、このような状況で使用される CPU は制限できます。デフォルト設定は、Agent が実行されているコアの使用率の 40% です。

もう 1 つの考慮事項は FIM ルールです。ファイルシステムが極端にビジー状態の場合、またはカスタムルールの適用範囲が広すぎる場合、監視対象のファイルやディレクトリごとに複数のイベントが発生され、CPU とメモリーがさらに消費される可能性があります。そのため、FIM ルールは、パフォーマンスが最適化されるように、できるだけ適用範囲を狭くする必要があります。

FIM パフォーマンスは、Linux 導入プロセスの一部として設定されるファイル監視制限でも低下する可能性があることに注意する必要があります。このような場合、制限を増やしてパフォーマンスを向上させることができます。Windows Server 側では、Agent が FIM 用のドライバを使用するので、メモリー使用量の上限は、カーネルメモリの 700MB に設定されています。

ネットワーク接続も Agent パフォーマンスに影響を与える可能性があります。Agent は通常、ログエントリを受け取り、JSON に変換し、各オブジェクト（私たちは「イベント」と呼んでいます）を F5 Distributed Cloud AIP バックエンドに送ります。Agent が持続的な接続を維持できない場合、イベントをローカルファイルにキャッシュしようとします。接続が再開されると、Agent は通常動作を継続しますが、キャッシュされたイベントを送信する分のオーバーヘッドが追加されます。

イベントの構造

以下の JSON 構造は、F5 Distributed Cloud AIP プラットフォームによって表面化された Linux ホストサーバーのイベントの例です。この例では、イベントはホスト OS 上の tsauditd によって送信され、バックエンドで処理されて、organization_id、time_id、その他の情報などの追加のメタデータで強化されていることを意味します。

例として、この Linux ホストサーバーのイベントは、バックエンドで処理されるときにアラートをトリガーします。これは、実行時でのカーネルモジュールの挿入は、疑わしい活動を示す強力な指標であるためですが、ほとんどのイベントは非常に日常的なものです。環境がうまく調整されていれば、アラートをトリガーするイベントデータは通常 1 パーセント程度です。

Linux Roots


```
{
       "event_type": "audit",
       "status": "success",
       "container_labels": null,
       "path": [
         "/usr/sbin/insmod",
         "/lib64/ld-linux-x86-64.so.2"
       ],
       "event_time": 1545224296000,
       "gid": 0,
       "_id": "c0c68b9a-038d-11e9-8ef5-0eb
9fbf2d436",
       "command": "insmod",
       "auid": 500,
       "container_image": null,
       "pid": 16708,
       "auser": "ec2-user",
       "organization_id": "xxxxx",
       "session": 2017,
       "exit": 0,
       "uid": 0,
       "cwd": "/home/ec2-user",
       "ppid": 16682,
       "args": [
         "insmod",
         "xpacket.ko"
       ],
       "tty": "pts0",
       "container_id": null,
       "syscall": "execve",
       "_insert_time": 1545224284685,
       "type": "start",
       "group": "ec2-user",
       "user": "root",
       "agent": {
         "name": "ip-xx-xx-xx-xx",
         "policy_id": null
       },
       "agent_id": "4f00dc5a-abca-11e8-bb29-e52888cc3b77",
       "is_agent_2": false,
       "exe": "/usr/bin/kmod",
       "arguments": "insmod xpacket.ko",
       "time_id": "c0dbe7a1-038d-11e9-ae98-7fc7b9401ba0"
}
```


注：スキーマは若干変更される場合があります。

図 1：Linux Roots イベント

以下の JSON 構造は、Distributed Cloud AIP プラットフォームによって表面化された Windows Server ホストのイベントの例です。ここでは、Windows Agent が Windows イベントログから WinSec イベントを取得し、処理とルールベースのアラートのためにバックエンドに送信しています。

Windows イベント


```
{
       "target_user": "Angela",
       "event_time": 1564159625008,
       "win_event_id": 4776,
       "src_host": "EC2AMAZ-CIF7K66",
       "workstation": "workstation",
       "auth_package": "MICROSOFT_AUTHENTICATION_PACKAGE_
V1_0",
       "status": "0xc0000064",
       "record_number": 7193237,
       "organization_id": "xxxxx",
       "session": 0,
       "_insert_time": 1564159626028,
       "_subtype": "Credential Validation",
       "summary": "The computer attempted to validate the
credentials for an account.",
       "event_type": "winsec",
       "agent": {
         "name": "EC2AMAZ-CIF7K66",
         "policy_id": null
       },
       "placement": "Event List"
}
```


図 2：Windows イベント

上記の Windows の例は、より日常的なクレデンシャル検証イベントです。そのままの場合、低重要度のアラートが作成されますが、ルールを調整すれば、アラートを生成するルールを Amazon EC2 タグで除外することで、この種のアラートを省略できます。

参考

ルール調整については本書の対象外ですが、大まかに説明すると、すべてのルールと処理は F5 Distributed Cloud AIP のバックエンドにあり、完全にカスタマイズ可能です。アラートルールは、PCI DSS、HIPAA、ISO 27001、およびその他の規制に対応するために、適切な監視と監査追跡が行われていることを証明するように設計されています。

F5 Distributed Cloud AIP は、他のデータも収集していますが、これも本書では対象外です。大まかに説明すると、Linux システムの脆弱性評価データは、ホストのパッケージマネージャから取得され、National Vulnerability Database と照合されます。Linux と Windows Server の両方において、F5 Distributed Cloud AIP は、EC2 リソースなどの AWS 固有のデータを取り込み、CloudTrail と統合して、特別に設計された一連のルールに基づき AWS サービス間の疑わしい API 相互作用に対してアラートを生成できます。

サポートされているオペレーティングシステム
F5 Distributed Cloud AIP は、現在、以下のオペレーティングシステムのさまざまなバージョンをサポートしています。

- Amazon Linux
- CentOS
- Debian
- Red Hat
- Ubuntu（Ubuntu 22.04 のサポートを含む）
- Windows Server

F5 Distributed Cloud AIP は、システムに導入されている Agent の自動更新を強制しません。

エージェントイベントに基づくアラートのユースケース

Linux Agent のイベントデータに基づく一連の一般的なセキュリティユースケースを文書化しています。ドキュメントサイトでは、アラートに関する詳細とコンテキストを提供していますが、以下に一般的なアラートシナリオの概要を示します。

1. ユーザーアクセスの監視

- あらゆるユーザーまたはグループの変更
- 特権の昇格、または未承認の sudo ユーザーによる試行
- 監視対象システムへの未許可の変更たとえば、規定の設定管理ツール以外で発生した手動による変更
- すべてのユーザーの TTY セッションの監査証跡
- root などの特権ユーザーアカウントの使用
- 異常なログインまたはユーザーアクセスの試行たとえば、パスワードの総当たり攻撃を示す可能性のある高い試行率

2. システム整合性監視

- ルートキットやマルウェアの存在を示唆する、カーネルモジュールの不正なロード
- 許可されたポートおよびサービスの使用における逸脱
- コマンド & コントロール活動を示す可能性のある新しいプロセス接続状態
- 異常なプロセス開始イベント
- 読み取り、転送、許可など、重要なファイルシステムの変更に関する監査証跡

3. ファイル整合性監視

- 重要な認証情報ファイルの変更またはアクセス
- 新しい実行可能ファイルやバイナリ交換のための特別なシステムディレクトリ (`/boot/` 、`/lib` 、`/usr/lib` 、`/bin` 、`/sbin` 、`/etc`) への変更
- 重要な設定ファイルへの未許可の変更
- 単純な「オープン」ファイルシステムイベントも含む、機密ファイルに対するデータ流出活動

4. ネットワーク活動監視

- 重要なシステムサービスの変更（NTP、DNS、Syslog）またはデーモンの再設定
- ウェブサーバー、データベースサーバーのバインドまたはプロキシノードのようなアプリケーションサービスの変更
- Telnet や FT など、システムアクセスのための安全でないプロトコルの使用

Threat Stack：F5 ソリューションになりました

Threat Stack は製品名称を変更し、F5 Distributed Cloud App Infrastructure Protection（AIP）となりました。本ソリューション、F5 のセキュリティオペレーションセンター（Distributed Cloud AIP Managed Security Services、Distributed Cloud AIP Insights を含む）、およびその他について詳しくは、F5 のクラウドセキュリティとコンプライアンスの専門家までお気軽にお問い合わせください。

クラウドセキュリティのお悩みは F5 のセキュリティ専門家にぜひお任せください。詳細およびデモのご予約については、F5 のウェブサイトをご覧ください。

[1] Red Hat Docs：セキュリティガイド「システム監査」
（https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/6/html/security_guide/chapsystem_auditing）

DC 12.2022 | JOB-CODE-WP-950855761-Agent