管理コンソール

イベント通知機能について

Version 2.1

FFRI Security, Inc.
株式会社FFRIセキュリティ

目次

著作権

免責事項

更新履歴

2010-11-14	1.0	パートナーサイト用
2014-04-03	1.1	更新
2014-10-09	1.2	・ 置換変数の出力例を追加
2015-04-24	1.3	FFR Enterprise Management Console 2.6 リリースに伴い資料を更新 ・ Static 分析（エンジン番号 9）を追加 ・ HIPS エンジンの$$_ActionCode_$$の説明を変更 ・ 脆弱性検出（エンジン番号 6）を追加 ・ 機械学習（エンジン番号 8）を追加
2015-10-26	1.4	FFR Enterprise Management Console 2.7 リリースに伴い資料を更新 ・ Windows Defender（エンジン番号 16）を追加 ・ HIPS エンジンの$$_DetectionReason_$$に 34-37 を追加
2016-09-20	1.5	FFR yarai 2.7.7 リリースに伴い資料を更新 ・ 脆弱性検出エンジンの$$_ActionCode_$$の説明を変更 ・ HIPS エンジンの$$_DetectionReason_$$に 38-39 を追加
2016-12-26	1.6	FFRI Enterprise Management Console 2.8 リリースに伴い資料を更新 ・ 製品名変更に伴い、文書中の製品名を更新 ・ HIPS エンジンの$$_DetectionReason_$$に 40-41 を追加 ・ スタティックエンジン分析(エンジン番号2番)の誤記修正(カンマ区切り→コロン区切り、仕様→使用に修正) ・「もくじ」を「目次」に変更
2017-04-05	1.7	FFRI Enterprise Management Console 2.9 リリースに伴い資料を更新 ・ HIPS エンジンの$$_DetectionReason_$$に 42、43 を追加 ・ HIPS エンジンの$$_DetectionReason_$$の 41 の意味を変更 ・ HIPS（エンジン番号 10）を追加
2018-07-19	1.8	FFRI AMC 3.2/FFRI Enterprise Management Console 2.12 リリースに伴い資料を更新 ・ HIPS エンジンの$$_DetectionReason_$$に 44 を追加

2019-10-18	1.9	FFRI Enterprise Management Console 2.13 リリースに伴い資料を更新 ・ HIPS エンジンの$$_DetectionReason_$$に 45 を追加
2019-11-20	1.10	FFRI AMC 3.3 リリースに伴い資料を更新 ・ 文書名を「管理コンソールイベント通知機能について」に変更 ・ 表紙にバージョン番号を記載 ・ Static 分析（エンジン番号 20）を追加
2020-06-11	1.11	社名と本社所在地を変更
2021-02-10	1.12	FFRI AMC 3.4.0/FFRI Enterprise Management Console 2.14 リリースに伴い資料を更新 ・ FFRI AMC 3.4 で追加した置換引数について追記 ・ Static 分析（エンジン番号 11）を追加 ・ ハッシュ検索（エンジン番号 32）を追加 ・ HIPS エンジンの$$_DetectionReason_$$に 46 を追加 ・ HIPS エンジンの$$_DetectionReason_$$の 33 の意味を変更 ・ 機械学習エンジンの$$_ActionCode_$$の誤記を修正
2021-09-15	1.13	FFRI AMC 3.4.6 リリースに伴い資料を更新 ・ FFRI AMC 3.4.6 で追加した過検出判定の置換引数 $$_SpecimenPath_$$を追記 ・ 隔離コマンド応答として実際には出力されないパターンの記載を削除
2022-04-12	1.14	FFRI AMC 3.5 リリースに伴い資料を更新 ・ Static 分析（エンジン番号 33）を追加 ・ 機械学習（エンジン番号 34）を追加 ・ HIPS エンジンの$$_DetectionReason_$$の 16、43 の $$_DetectionReasonString_$$を変更 ・ HIPS エンジンの$$_DetectionReason_$$に 47-48 を追加
2023-01-27	2.0	・ サポート終了に伴い FFRI Enterprise Management Console の記載を削除 ・ 更新履歴の記載を修正（対応する製品バージョンを記載、誤記修正）
2024-04-22	2.1	FFRI AMC 3.6 リリースに伴い資料を更新 ・ 検出関連通知で利用可能な置換引数に IOA レコードを追加 ・ HIPS（エンジン番号 12,13）を追加 ・ Static 分析（エンジン番号 35,36,37）を追加 ・ Static 分析エンジン、Sandbox エンジンの

$$_ActionCode_$$に 6~9 を追加

- HIPS エンジンの$$_ActionCode_$$の 4 の文言変更および 5 の追加
- 検知理由に IOA レコード通知を追加
- HIPS エンジンの$$_DetectionReason_$$に 49-50 を追加
- 駆除状況の 9、11 の文言を変更、12 の追加
- 過検出判定状況の $$_DetectionReviewStatus_$$ に 9 (検体ファイルエラー) を追加

文書情報

発行元: 株式会社FFRIセキュリティ

連絡先: 株式会社FFRIセキュリティ

sales@ffri.jp

〒100-0005

東京都千代田区丸の内3丁目3番1号 新東京ビル2階

1. 管理コンソールにおけるイベント通知機能

外部システムと連携するために、管理コンソールにはイベント通知機能があります。この機能により、管理コンソールがイベントを受信したタイミングで任意の外部のプログラムを呼び出し、様々な検出パラメータを渡すことができます。その結果、イベントを受信したタイミングで SNMP トラップを上げる、社内で使われている独自の監視システムと連携する、といったカスタマイズが可能になります。

2. 設定画面

3.4 以降の管理コンソール[1]の「通知プログラム設定」において「新規作成」を選択すると以下の画面が表示されます。この画面でプログラムのパスを指定し、引数を設定し「プログラムを追加」ボタンを押下することで、通知プログラムを登録することができます。

[1] AMC 3.3 以前では FFRI Enterprise Management Concole 同様、検出時の通知のみ利用可能でした。

3. イベント通知機能設定項目

項目名	内容	備考
組織名	このプログラムを使用する組織名を指定して下さい。	組織が統括管理組織のみの場合は設定しません。
設定名	この起動設定に名前を付けることが可能です。	
実行ファイルパス	起動するプログラムのパスを指定して下さい。	実際に存在しないパスは入力エラーとなります。
引数	プログラムの引数として渡す文字列を設定することができます。 スペース区切りで任意の引数を渡すことが可能です。 引数には「$$_?????_$$」という特殊な形式を指定することができます。これは置換引数と呼ばれ、管理コンソールがイベントを受信して通知プログラムを起動する時に、検出時刻や検出エンジンといったパラメータに置き換わります。	利用可能な置換引数は通知種類により異なります。 上の通知種類のリストから通知対象としたい通知内容を選択することで、その通知種類で使用可能な置換引数が下のリストで選択可能となります。 下のリストの要素をクリックすることで引数欄へのキーボード入力を省略することができます。

4. 置換引数 一覧

利用可能な置換引数の一覧を示します。

全通知共通で利用可能な置換引数

下記の置換引数は全ての通知種類で利用可能です。

置換引数		内容
通知種類名	$$_AlertItem_$$	検出、過検出判定(判定完了)など
組織名	$$_OrganizationName_$$	関連する組織名
送信先メールアドレス	$$_ToMailAddresses_$$	メールアドレスの設定がある場合、そのメールアドレス

クライアントイベント通知で利用可能な置換引数

下記の置換引数はクライアントにて発生するイベントに関する通知種類で利用可能です。そのイベントが発生したクライアントを確認するための情報を提供します。

置換引数		内容
ドメイン名	$$_DomainName_$$	所属するドメイン名
ホスト名	$$_HostName_$$	クライアントのホスト名
グループ名	$$_GroupName_$$	所属するグループ名
IP アドレス	$$_IPAddress_$$	クライアントの IP アドレス 取得不可の場合「0.0.0.0」となる場合があります。
プロダクトバージョン	$$_ProductVersion_$$	FFRI yarai のバージョン
エンジンバージョン	$$_EngineVersion_$$	FFRI yarai のバージョン
GUI バージョン	$$_GUIVersion_$$	FFRI yarai のバージョン
データバージョン	$$_DataVersion_$$	FFRI yarai のバージョン

置換引数		内容
OS	$$_OS_$$	オペレーティングシステム (例:Windows 10)
前回アップデートした日時	$$_LastUpdate_$$	そのホストの最終アップデート日時[2]
MAC アドレス	$$_MacAddress_$$	クライアントの MAC アドレス
クライアント端末識別子	$$_ClientKey_$$	クライアント端末識別子

後述「6 イベント通知設定例」のスクリプトを設定した AMC での変換例を以下に示します。

```
組織名: organization_of_administrator
グループ名: WORKGROUP
ドメイン名: WORKGROUP
ホスト名: WIN-U9C4SGT5BU8
IP アドレス: 192.168.42.129
プロダクトバージョン: 3.4.685.0
エンジンバージョン: 3.4.685.0
GUI バージョン: 3.4.685.0
データバージョン: 3.4.685.0
OS(オペレーティングシステム): Windows 7
最終アップデート: 2020/06/01 11:57:28
ライセンス状況: 無効
```

検出関連通知で利用可能な置換引数

検出関連通知とは以下の通知種類を指します。検出通知以外は AMC でのみ設定可能です。各通知種類についての説明は AMC のマニュアルを参照ください。

- 検出
- Windows Defender 検出
- EDR 脅威(手動配布ジョブ)
- EDR 脅威(自動作成ジョブ)

[2] AMC 3.4 より「2020/06/01 11:57:28」というフォーマットになります。AMC 3.3 以前のバージョンでは「2020-06-01 11:57:28」でした。

- 過検出判定(例外リスト登録済み)
- 過検出判定(判定完了)
- 過検出判定(その他)
- IOA レコード

下記の置換引数は検出に関連する通知種類で利用可能です。検出機構や検出理由に関する情報を提供します。

置換引数		内容
検出日時	$$_DetectionDate_$$	検出した日時
検出したプロセス	$$_DetectionProcess_$$	検出したファイルパス
ハッシュ（SHA-256）	$$_Hash_$$	検出したファイルのハッシュ値
検出機構	$$_DetectionEngine_$$	詳細リンク: 検出機構
	$$_DetectionEngineString_$$	
検出理由	$$_DetectionReason_$$	詳細リンク: 検知理由
	$$_DetectionReasonString_$$	
検出時の動作	$$_ActionCode_$$	詳細リンク: 検出時の動作
	$$_ActionCodeString_$$	
駆除状況	$$_Extermination_$$	詳細リンク: 駆除状況
	$$_ExterminationString_$$	
過検出判定状況	$$_DetectionReviewStatus_$$	詳細リンク: 過検出判定状況
	$$_DetectionReviewStatusString_$$	
過検出更新日時	$$_DetectionReviewDate_$$	過検出判定状況が更新された日時
検体の詳細情報	$$_SpecimenDetailInfo_$$	検体の詳細情報を表す JSON データ
MITRE ATT&CK 情報	$$_MitreAttack_$$	MITRE ATT&CK 情報を表す JSON データ
IOA レコード	$$_IOA_$$	IOA(Indicator Of Attack)を表す JSON データ

Static 分析エンジンで検出した際の検出通知例を以下に示します。

```
検出日時: 2020/06/19 10:58:51
検出したプロセス: C:¥test¥yarai_StaticTest_AntiSecurity.exe
ハッシュ値: 3b53934eb07d8452c2ab7e212d7cf214bdd8f1800be8afc640dbb0aa73443022
検出機構: Static 分析 (Code: 2)
検出時の動作:  (Code: 0)
検出理由: セキュリティソフトウェアに対して攻撃を行う可能性があります。 (Code: 0:0:0:0:0:1:0:0)
駆除ステータス: 駆除されていない (Code: 0)
過検出判定状況: 未判定 (Code: 0)
```

HIPS エンジンで検出した際の同じ通知プログラムでの検出例を以下に示します。

```
検出日時: 2020/06/26 12:14:45
検出したプロセス: C:¥Users¥takuro.kitamura¥AppData¥Local¥Temp¥testmalware.exe
ハッシュ値: d1031f20595e59c47116b53c95ee9c828ae544f3ca2572da98c897e592906931
検出機構: HIPS (Code: 4)
検出時の動作: プロセスを終了させました。 (Code: 3)
検出理由: プロセスが自分自身のコピーを実行しようとしました。 (Code: 11)
駆除ステータス: 駆除されていない (Code: 0)
過検出判定状況: 未判定 (Code: 0)
```

EDR 脅威(手動配布ジョブ)・EDR 脅威(自動作成ジョブ)で利用可能な置換引数

EDR 脅威については検出関連の置換引数に加えて以下の置換引数を利用することができます。

置換引数		内容
ジョブ名	$$_HuntingJob_$$	配布したジョブ名
IOC グループ名	$$_IocGroup_$$	ファイルを検出した際にその検出したファイル情報が含まれる IOC グループ名
URL	$$_TaxiiServerAddress_$$	DISCOVERY リクエストの送信先 TAXII サーバーの URL[3]
コレクション名	$$_TaxiiFeedName_$$	TAXII サーバーから取得した FEED の名前[3]

[3] EDR 脅威(自動作成ジョブ)でのみ置換される置換引数となります。

過検出判定（例外リスト登録済み/判定完了/その他）で利用可能な置換引数

過検出判定については検出関連の置換引数に加えて以下の置換引数を利用することができます。

置換引数		内容
検体保管パス	$$_SpecimenPath_$$	クライアントから収集した検体 zip ファイルの管理コンソール上での保管パス

スキャンエンジン停止・ライセンス期限切れで利用可能な置換引数

置換引数		内容
ライセンス有効/無効	$$_LicenseStatus_$$	有効 or 無効

隔離実施結果で利用可能な置換引数

置換引数		内容
隔離状況	$$_IsolationStatus_$$	解除状態 or 隔離状態
隔離コマンド応答	$$_IsolationResponse_$$	詳細リンク: 隔離コマンド応答
隔離実施トリガー	$$_IsolationTrigger_$$	自動隔離 or 手動配布

配布イベント通知で利用可能な置換引数

下記の置換引数は配布イベントに関連する通知種類で共通して利用可能です。配布物名や進捗状況に関する情報を提供します。

置換引数		内容
配布物	$$_Name_$$	配布物名(ex.ポリシー名、シリアルナンバー)
配布日時	$$_StartTime_$$	AMC で配布を行った時刻
配布ステータス	$$_Status_$$	配布完了 / 接続待ち
対象クライアント数	$$_TotalClients_$$	配布クライアント総数
配布完了	$$_FinishClients_$$	配布を行い成功または失敗を応答したクライアントの総数
成功クライアント数	$$_SuccessClients_$$	成功を応答したクライアントの総数
失敗クライアント数	$$_FailedClients_$$	失敗を応答したクライアントの総数
対象外クライアント数	$$_ExcludedClients_$$	配布は行ったがポリシーで無効となった場合など、対象外となったクライアントの総数

アップデート配布時の配布通知例を以下に示します。

```
配布物：アップデート配布進捗
名称: 3.4.651.0
配布日時: 2020-04-24 21:27:29.40063+09

配布ステータス: 接続待ち
対象クライアント数: 100
完了クライアント数: 30
正常クライアント数: 20
失敗クライアント数: 10
対象外クライアント数: 0
```

AMC エラーイベント通知で利用可能な置換引数

下記の置換引数は AMC エラーイベントに関連する通知種類で利用可能です。エラー発生日時やその内容に関する情報を提供します。

置換引数		内容
失敗回数	$$_ErrorCount_$$	この通知に含まれるエラー件数を表します
エラー発生日時	$$_ErrorDate_$$	エラー発生日時をカンマ区切りで複数出力します。
エラーコード	$$_ErrorCode_$$	エラーのコードをカンマ区切りで複数出力します。

サーバーアクセス失敗通知で利用可能な置換引数

置換引数		内容
アクセスサーバー名	$$_ServerAddress_$$	アクセスに失敗した URL を表します。

通知プログラム起動失敗通知で利用可能な置換引数

置換引数		内容
通知プログラム名	$$_ProgramName_$$	起動に失敗した通知プログラムの設定名を出力します。
通知プログラム引数	$$_Args_$$	起動に失敗した通知プログラムに実際に渡された置換後の引数を表します。

リソース通知で利用可能な置換引数

下記の置換引数はリソースに関連する通知種類で利用可能です。リソースの現在使用量と閾値に関する情報を提供します。

置換引数		内容
通知種類	$$_AlertStatus_$$	「アラート閾値を超過しています。」or「アラート閾値以下に回復しました。」
現在使用量	$$_ResourceQuantity_$$	現在の使用量
最大値	$$_ResourceMax_$$	閾値として設定した最大量
発生日時	$$_EventDate_$$	この通知の発生日時

5. 置換引数 詳細

コードと文言[4]で構成される置換引数について詳細を示します。

検出機構

脅威を検出したエンジン、もしくはイベントの種類を表します。

置換引数$$_DetectionEngine_$$はコード（数字）に、$$_DetectionEngineString_$$は対応する文言に置き換わります。

コード $$_DetectionEngine_$$	文言 $$_DetectionEngineString_$$	説明
1	Static 分析	Static 分析エンジン(マルウェアとして定義されています)
2	Static 分析	Static 分析エンジン
3	Sandbox	Sandbox エンジン
4	HIPS	HIPS エンジン
5	ZDP	ZDP エンジン
6	ZDP	ZDP エンジン（設計脆弱性）
7	Static 分析	Static 分析エンジン（.NET マルウェア）
8	機械学習	機械学習エンジン
9	Static 分析	Static 分析エンジン（VisualBasic マルウェア）
10	HIPS	HIPS エンジン（悪用検知） Powershell など正規のプログラムを悪用されたことを検出したケースです。
11	Static 分析	Static 分析エンジン（スタティック機械学習）
12	HIPS	HIPS エンジン（複製された正規プログラムの悪用検出）
13	HIPS	HIPS エンジン（不正停止検出機能による検出）

[4] 管理コンソールのバージョンアップの際に文言内容が変更されるケースがあります。プログラム内部にて文言内容を用いた判定などは行わないようにして頂くことを推奨します。

コード $$_DetectionEngine_$$	文言 $$_DetectionEngineString_$$	説明
16	Defender	Windows Defender（検出）
20	Static 分析	Static 分析エンジン(クラウド連携機能による検出)
32	ハッシュ検索	EDR ハッシュ検索
33	Static 分析	Static 分析エンジン（オンデマンドスキャンでのディープマクロ分析）
34	機械学習エンジン	3.5 以降のクライアントにおける機械学習エンジン
35	Static 分析	Static 分析エンジン（オンデマンドスキャンでのディープマクロ分析　マクロ分析による検出）
36	Static 分析	Static 分析エンジン（オンデマンドスキャンでのディープマクロ分析　ファイルハッシュシグネチャ DB による検出）
37	Static 分析	Static 分析エンジン（オンデマンドスキャンでのディープマクロ分析　マクロハッシュシグネチャ DB による検出）

検出時の動作

脅威検出時に FFRI yarai が取ったアクションを示す情報に置き換わります。

置換引数$$_ActionCode_$$はコード（数字）に、$$_ActionCodeString_$$は対応する文言に置き換わります。

コードは検出機構($$_DetectionEngine_$$の数字)毎に異なる意味を表します。

Static 分析エンジン、Sandbox エンジンの場合

コード $$_ActionCode_$$	文言 $$_ActionCodeString_$$	説明
6	オンデマンドスキャンしたファイルをマルウェアとして分類しました。	オンデマンドスキャンされたときにマルウェアとして検出しました
7	作成されたファイルをマルウェアとして分類	ファイルが生成されたときにマル

	しました。	ウェアとして検出しました
8	ファイルの実行を阻止しました。	ファイルの実行を阻止しました
9	ファイルの実行を許可しました。	ログを出力してファイルの実行を許可しました

※検出機構コードは、1、2、3、7、9、11、20、33、34、35、36、37

HIPS エンジン(検出機構コード 4、10、12、13)の場合

コード $$_ActionCode_$$	文言 $$_ActionCodeString_$$	説明
1	脅威を緩和しました。	脅威を軽減させた上で実行を継続させました
2	処理の実行を拒否しました。	不審な挙動をブロックしました
3	プロセスを終了させました。	対象プロセスを停止させました
4	処理の実行を許可しました。	脅威を検出し通知のみ実施しました
5	IOA レコードを送信して処理の実行を許可しました。	脅威となりうる振る舞い(IOA)を検出し通知のみ実施しました

機械学習エンジン(検出機構コード 8)の場合

コード $$_ActionCode_$$	文言 $$_ActionCodeString_$$	説明
1	脅威を緩和しました。	脅威を軽減させた上で実行を継続させました
2	処理の実行を拒否しました。	不審な挙動をブロックしました
3	プロセスを終了させました。	対象プロセスを停止させました
4	処理を許可しました。(検知時の動作: ログ出力)	脅威を検出し通知のみ実施しました

ZDP エンジン(検出機構コード 5)の場合

コード $$_ActionCode_$$	文言 $$_ActionCodeString_$$	説明
3	処理の実行を許可しました。	脅威を検出し通知のみ実施しました
4	プロセスを終了させました。	対象プロセスを停止させました

その他の場合

コード $$_ActionCode_$$	文言 $$_ActionCodeString_$$	説明
0	スペース	コード 0 に対応する文言は有意な文字列は出力されず、スペースが出力されます。

検知理由

検出理由を示す特殊な文字列または数字に置き換わります。検出エンジン毎に異なる形式を取ります。

Static 分析エンジン(検出機構コード 2)の場合

静的解析による解析結果を示す「1:3:1:0:0:0:0:0」というフォーマットの文字列に置き換わります。
置換引数$$_DetectionReason_$$は特殊な文字列に、$$_DetectionReasonString_$$は対応する文言に置き換わります。

コード $$_DetectionReason_$$	文言 $$_ DetectionReasonString_$$
1:3:1:0:0:0:0:0	0 でない部分の文言がスペース区切りで列挙されます。

コロン区切りでそれぞれのフィールドは下記のような意味を持ちます。

引数の場所	意味	文言 $$_ DetectionReasonString_$$
第 1 引数	異常なコードセクションがあれば 1 に、そうでなければ 0 になります。	コードセクションの構造に異常を検出しました。
第 2 引数	1 であればパッカー、2 であればリソース内に実行ファイルが埋め込まれており、未知のパッカーであれば 3 に、それ以外であれば 0 になります。	「パッカーを検出しました。悪意あるコードが隠蔽されている可能性があります。」 など
第 3 引数	未知の怪しいセクションがあれば 1 に、それ以外は 0 になります。	不審なセクション構造を検出しました。
第 4 引数	マルウェアが使うセクションがあれば 1 に、それ以外は 0 になります。	マルウェア特有のセクションを検出しました。
第 5 引数	マルウェアが使うアイコンを使用していれば 1 に、それ以外は 0 になります。	偽装された実行ファイルを検出しました。
第 6 引数	AntiVirus や Firewall を見付けて無効にしようとする痕跡があると 1 に、それ以外は 0 になります。	セキュリティソフトウェアに対して攻撃を行う可能性があります。
第 7 引数	IRC 等の怪しいネットワークアクセスをしようとする痕跡があると 1 に、それ以外は 0 になります。	不審なネットワークアクセスを行う可能性があります。
第 8 引数	その他の怪しい挙動をする痕跡があると 1 に、それ以外は 0 になります。	マルウェア特有の行動を取る可能性があります。

HIPS エンジン(検出機構コード 4、10、12、13)の場合

検出理由を示す数字に置き換わります。

コード $$_DetectionReason_$$	文言 $$_ DetectionReasonString_$$
1	プロセスがデバッグ権限を取得しようとしました。
2	プロセスが別のプロセスを不正利用しようとしました。

コード $$_DetectionReason_$$	文言 $$_ DetectionReasonString_$$
3	プロセスが別のプロセスのスレッドを不正利用しようとしました。
4	プロセスが自分自身の実行ファイルを操作して痕跡を消そうとしている疑いがあります。
5	プロセスが頻繁に SMTP 接続を確立していました。
6	プロセスが大量に不信な SMTP 応答を受信していました。
7	プロセスが頻繁に SMTP コマンドを送信していました。
8	他のプロセスのメモリをアンマップしようとしました。
9	偽装したシステムファイルを生成しようとしました。
10	呼び出し元で適切に解決されていない API が呼び出されました。
11	プロセスが自分自身のコピーを実行しようとしました。
12	隠しプロセスがシステムユーティリティを実行しようとしました。
13	隠しプロセスが自身で生成したバッチファイルを実行しようとしました。
14	隠しプロセスがムービーをキャプチャーしようとしました。
15	プロセスが攻撃プログラムを起動しようとしています。
16	アプリケーションが不審なサービスを登録しようとしました。
17	プロセスがキーストロークを監視しようとしました。
18	隠しプロセスが Windows フックをインストールしようとしました。スパイウェアに利用される可能性があります。
19	プロセスが ZwLoadDriver を呼び出そうとしました。
20	プロセスが書き込み権限で「¥¥Device¥¥PhysicalMemory」を開こうとしました。
21	プロセスがバックドアを設置して、ドライバを読み込もうとしました。
22	アプリケーションが自身を別のプロセスに挿入しようとしました。
23	アプリケーションが不審な方法でファイルをコピーしようとしました。
24	アプリケーションが自分自身のプログラムファイルを消そうとして、痕跡を消そうとしています。
25	アプリケーションが自分自身のプログラムファイルを書き換えようとしています。

コード $$_DetectionReason_$$	文言 $$_ DetectionReasonString_$$
26	アプリケーションに異常な解析対策が施されています。
27	アプリケーションが他のプロセスに侵入しようとしています。
28	プロセスが HOSTS ファイルを不正に変更し、重要なアップデートなどを無効にしようとしました。
29	プロセスがドライブの自動再生で実行ファイルを登録しようとしました。
30	隠しプロセスがシステムのファイアウォールを無効にするよう、レジストリを変更しようとしました。
31	プロセスが他のプロセスをデバッグできるよう、レジストリを変更しようとしました。
32	隠しプロセスが Windows 起動時に自身が実行されるよう、レジストリを変更しようとしました。
33	隠しプロセスがログオンの設定に関する不審なレジストリ設定を行いました。
34	不審なファイル削除を検知しました。
35	不審なファイル生成を検知しました。
36	不審なプロセス生成を検知しました。
37	不審なメモリ書換えを検知しました。
38	物理ドライブを開こうとしました。
39	レジストリに悪意あるコードを書き込もうとしました。
40	プロセスが不審なファイル検索を試みました。
41	不審なマクロまたはスクリプトを検知しました。
42	自己解凍書庫による不審なプロセス生成を検知しました。
43	ファイルレスマルウェアの挙動を検出しました。
44	プロセスがクレデンシャル情報への不審なアクセスを試みました。
45	不審な DLL 読み込みを検知しました。
46	不審な権限昇格を検知しました。
47	複製された正規プログラムの悪用を検出しました。
48	不審なマクロの構造を検出しました。

コード $$_DetectionReason_$$	文言 $$_ DetectionReasonString_$$
49	不正停止検出機能により、製品の重要プロセスを強制停止から保護しました。
50	不正停止検出機能により、製品の重要プロセスを強制停止から保護しました。

Windows Defender 検出(検出機構コード 16)の場合

コード $$_DetectionReason_$$	文言 $$_ DetectionReasonString_$$	説明
0	Trojan:Win32/Ymacco.AA6B Program:Win32/Vigram.A など	検出理由コードは 0 固定となりますが、検出理由の文言は Windows Defender が付与したマルウェア名となります。

IOA レコード通知の場合

コード $$_DetectionReason_$$	文言 $$_ DetectionReasonString_$$
1	ファイルを開きました。
2	プロセスが開始しました。
3	DLL が読み込まれました。
4	ファイルをコピーしました。
6	コマンドを実行しました。
7	レジストリに書き込みました。
8	通信しました。
10	イベントログを削除しました。
18	WMI イベントを登録しました。
20	他プロセスのメモリを読み込みました。
26	特殊な方法でプロセスを起動しました。

コード $$_DetectionReason_$$	文言 $$_ DetectionReasonString_$$
33	ユーザー認証を試行しました。

IOA レコード機能で通知される JSON には、IOA レコードの他に「検体の詳細情報」および「MITRE ATT&CK」の情報が含まれる場合があります。

IOA レコードと検体の詳細情報については、別紙「イベント通知機能で送信される情報の詳細」をご参照ください。

MITRE ATT&CK の情報には以下が含まれます。

MITRE ATT&CK

- MITRE ATT&CK のバージョン
- テクニック ID

その他の場合

コード $$_DetectionReason_$$	文言 $$_ DetectionReasonString_$$	説明
0	エンジンコードに応じた文言	常にコードは 0 となり、文言は yarai ログ画面で確認できる文字列となります。

駆除状況

置換引数$$_Extermination_$$はコード（数字）に、$$_ExterminationString_$$は対応する文言に置き換わります。

コード $$_Extermination_$$	文言 $$_ExterminationString_$$	説明
0	駆除されていない	駆除を行っていない状態です。
1	接続待ち（駆除）	駆除コマンドを配布するために当該クライアントが接続するのを待っている状態です。
2	駆除中	クライアントに駆除コマンドを送信した後、クライアントか

コード $$_Extermination_$$	文言 $$_ExterminationString_$$	説明
		らのコマンドの応答を受信するまでの間の状態です。
3	駆除失敗	駆除に失敗した状態です。
4	不完全な駆除（駆除失敗詳細あり）	マルウェア本体の駆除には成功したものの、マルウェアが操作したファイルやレジストリを復元できなかった状態です。
5	接続待ち（駆除失敗詳細の削除）	「駆除失敗の詳細」削除コマンドを配布するために当該クライアントが接続するのを待っている状態です。
6		
7	駆除失敗詳細の削除中	クライアントに「駆除失敗の詳細」削除コマンドを送信した後、クライアントからのコマンドの応答を受信するまでの間の状態です。
8	駆除失敗詳細の削除失敗	「駆除失敗の詳細」の削除に失敗した状態です。
9	対象外(ファイルレスマルウェアの検出)	PowerShell 等の正常なスクリプト等が悪用された検出の場合にこの状態となります。「駆除」をしても、クライアントに駆除コマンドを配布することはありません。
10	駆除済み	駆除が完了している状態です。
11	対象外(ZDP エンジンによる検出)	脆弱性攻撃を ZDP エンジンにて検出した場合にこの状態となります。「駆除」をしても、クライアントに駆除コマンドを配布することはありません。
12	対象外(不正停止検出機能による検出)	不正停止検出機能による検出の場合にこの状態となります。「駆除」をしても、クライアントに駆除コマンドを配布することはありません。
16	管理コンソール駆除対象外	Windows Defender で検出した場合にこの状態となります。管理コンソールから本検出対象の「駆除」を配布することはできません。

過検出判定状況

置換引数$$_DetectionReviewStatus_$$はコード(数字)に、$$_DetectionReviewStatusString_$$は対応する文言に置き換わります。

コード $$_DetectionReviewStatus_$$	文言 $$_DetectionReviewStatusString_$$	説明
0	未判定	過検出判定を行っていません。
1	判定依頼中	過検出判定システムへの問い合わせを手動で行った後、実際に問い合わせるまでの間の状態を表します。
2	例外リスト登録済み	過検出判定の結果にてマルウェアでは無いと判定され例外リストに登録されました。
3	判定完了	過検出判定の結果、マルウェアの疑いがあると判定されました。ご質問等がある場合はサポート窓口へお問い合わせください。(ただし、内容によりましては回答にお時間を頂戴すること、回答しかねる場合がございますので、あらかじめご了承ください。)
4	照合なし	過検出判定システムに照合対象となるマルウェアの情報が存在しない状況です。 判定を進めるには検体のアップロードが必要となりますが、管理コンソールの「過検出判定設定」-「FFRIセキュリティに検体を提供する」が「無効」となっている場合にはこの状態のままとなります。
5	検体アップロード待ち	過検出判定システムに照合対象となるマルウェアの情報が存在しなかった場合、判定を進めるには検体のアップロードが必要となりますが FFRI yarai に配布されたポリシーの設定にて「検体収集」-「検体収集の使用」を「OFF」にしている場合など FFRI yarai から検体の収集が行えない場合にはこの状態

コード $$_DetectionReviewStatus_$$	文言 $$_DetectionReviewStatusString_$$	説明
		となります。
6	解析中	検体が自動アップロードされFFRIセキュリティによる過検出判定結果が確定するまでの間の状態です。
7	対象外	過検出判定システムへの問い合わせ対象外となる検出機構によるマルウェア情報であることを表します。EDR ハッシュ検索、HIPS エンジン(悪用検知)、ZDP エンジン、ZDP エンジン(設計脆弱性)による検知が対象外となります。
8	データが不十分	FFRI yarai に配布されたポリシーの設定にて「検体収集」-「検体収集対象」を「ログのみ」にしている場合やハッシュ値が空白である場合など、FFRI yarai から過検出判定に必要な検体収集が行えていない状況です。
9	検体ファイルエラー	過検出判定システムへ検体がアップロードされる際、検体ファイルに何らかの異常があり、情報を取得できず、過検出判定システムへと検体がアップロードできていない状態です。

隔離コマンド応答

クライアントに隔離を配布した結果を示す文言に置き換わります。

文言 $$_IsolationResponse_$$	説明
成功	配布に成功しました。
ポリシー無効	隔離が有効なポリシーが配布されておらず、隔離ができませんでした。

文言 $$_IsolationResponse_$$	説明
エラー応答(数値)	上記以外の理由で隔離に失敗しました。

6. イベント通知設定例

以下のようなバッチファイルを作成、通知プログラムとして設定した時の出力例を示します。

```
set ECHOFILE="C:¥FFRIAlertTest¥echofile.log"

echo ------------ >> %ECHOFILE%
echo Alert Test! >> %ECHOFILE%
echo %1          >> %ECHOFILE%
echo %2          >> %ECHOFILE%
echo %3          >> %ECHOFILE%
echo %4          >> %ECHOFILE%
echo %5          >> %ECHOFILE%
echo %6          >> %ECHOFILE%
echo %7          >> %ECHOFILE%
echo %8          >> %ECHOFILE%
echo %9          >> %ECHOFILE%
echo ------------ >> %ECHOFILE%
```

ここでは、予め c:¥FFRIAlertTest フォルダが存在し、引数としては以下のように指定したものとします。

```
$$_ActionCode_$$ $$_DetectionDate_$$ $$_DetectionEngine_$$ $$_DetectionProcess_$$ $
$_DetectionReason_$$ $$_DomainName_$$ $$_HostName_$$ $$_IPAddress_$$ $$_ProductVers
ion_$$
```

その結果の echofile.log の出力例は以下のようになります。

```
------------
Alert Test!
0
"2009/12/16 05:53:37"
2
"C:¥Program Files¥Cain¥Cain.exe"
1:3:1:0:0:0:0:0
FFRI Security
WS-KANAI
10.0.1.130
1.1.485.0
------------
```

初めのイベントは、Static 分析エンジン(2)による検出です。Cain.exe には異常なコードセクションがあり、未知のパッカーで、さらに未知のセクションもあります。FFRI yarai のアクションとしては、脅威をブロック(0)しました。

```
-----------
Alert Test!
4
"2009/12/16 05:56:00"
5
"C:¥ Program Files¥bin¥bof.exe"
0
FFRI Security
WS-KANAI
10.0.1.130
1.1.485.0
-----------
```

二つ目のイベントは、ZDP エンジン(5)による脆弱性検出です。FFRI yarai のアクションとしては、対象プロセスを停止(4)しました。

```
-----------
Alert Test!
2
"2009/12/16 06:43:31"
4
"C:¥ Program Files¥HIPS¥hipstest.exe"
19
FFRI Security
WS-KANAI
10.0.1.130
1.1.485.0
-----------
```

三番目のイベントは HIPS エンジン(4)による検出です。「プロセスが ZwLoadDriver を呼び出そうとしたため(19)に HIPS が検出しました。FFRI yarai のアクションとしては、この脅威を無視(2)しました。

また「クライアントイベント通知で利用可能な置換引数」のためのスクリプト設定例を以下に示します。

```
set ECHOFILE="C:¥FFRIAlertTest¥echofile.log"

組織名: %1>                              %ECHOFILE%
グループ名: %2>>                         %ECHOFILE%
ドメイン名: %3>>                         %ECHOFILE%
ホスト名: %4>>                           %ECHOFILE%
IP アドレス: %5>>                        %ECHOFILE%
プロダクトバージョン: %6>>               %ECHOFILE%
OS(オペレーティングシステム): %7>> %ECHOFILE%
最終アップデート: %8>>                   %ECHOFILE%
ライセンス状況: %9>>                     %ECHOFILE%
```

この時の引数としては以下のように指定します。

```
$$_OrganizationName_$$ $$_GroupName_$$ $$_DomainName_$$ $$_HostName_$$ $$_IPAddress
_$$ $$_ProductVersion_$$ $$_OS_$$ $$_LastUpdate_$$ $$_LicenseStatus_$$
```