

Internet Infrastructure Review

Vol.25

November
2014

インフラストラクチャセキュリティ

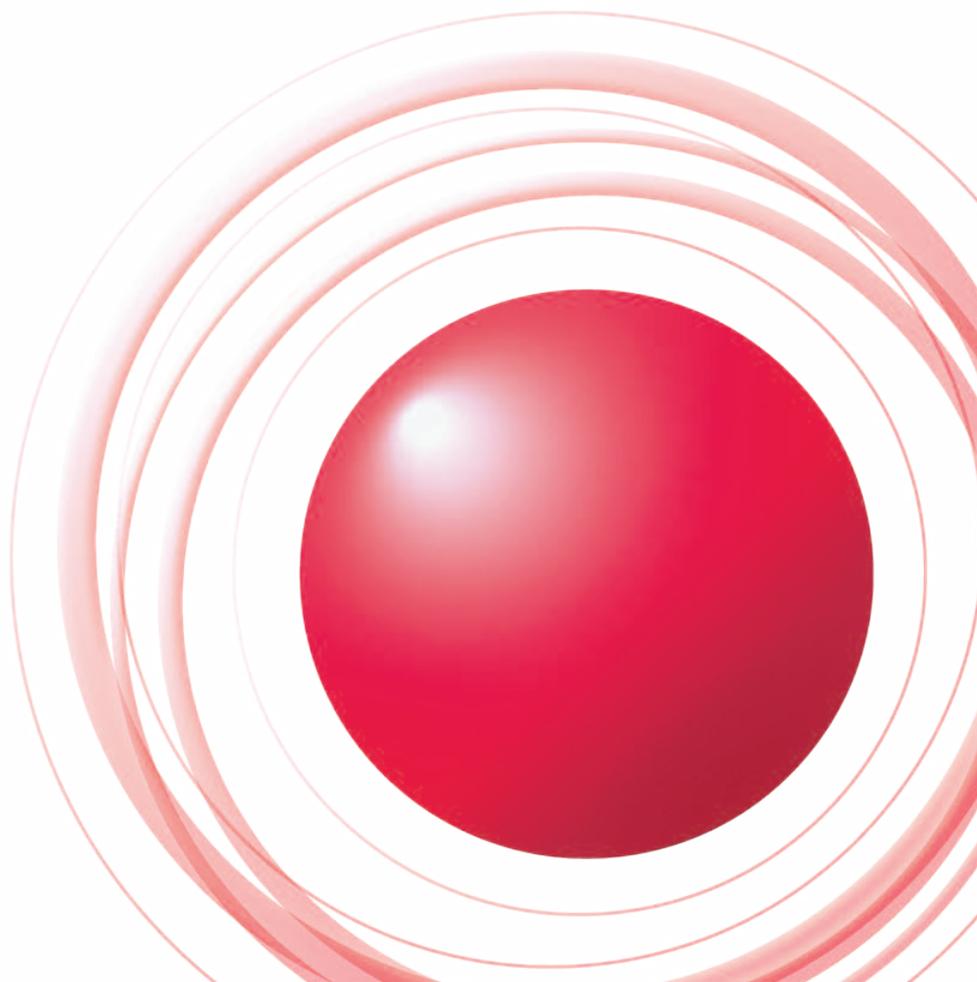
Bashの脆弱性Shellshockについて

コンテンツ配信

最新ストリーミング技術

クラウドコンピューティングテクノロジー

SDNソフトウェアスイッチ「Lagopus」



エグゼクティブサマリ	3
1. インフラストラクチャセキュリティ	4
1.1 はじめに	4
1.2 インシデントサマリ	4
1.3 インシデントサーベイ	11
1.3.1 DDoS攻撃	11
1.3.2 マルウェアの活動	13
1.3.3 SQLインジェクション攻撃	16
1.3.4 Webサイト改ざん	17
1.4 フォーカスリサーチ	18
1.4.1 Bashの脆弱性Shellshockについて	18
1.4.2 POODLE attack	21
1.4.3 リスト型攻撃の発生状況と対策	25
1.5 おわりに	27
2. コンテンツ配信	28
2.1 大規模配信	28
2.2 モバイルの話	29
2.3 H.265の登場と4K	29
2.4 4K	30
2.5 MPEG-DASH	30
2.6 甲子園実例	32
2.7 おわりに	33
3. クラウドコンピューティングテクノロジー	34
3.1 SDNとOpenFlow	34
3.2 OpenFlow	35
3.3 ソフトウェアスイッチ	36
3.4 SDNソフトウェアスイッチLagopus	36
3.5 ソフトウェアスイッチのボトルネック	37
3.5.1 受信割り込み、送信完了割り込み処理	37
3.5.2 パケットのメモリコピー	37
3.5.3 OSによるプロセススケジューリング	37
3.5.4 メモリブロックの確保と解放、ページフォルト	37
3.5.5 リソースへの排他アクセス	37
3.6 ボトルネックを解消するIntel DPDK	38
3.7 Lagopusにおける性能向上策	38
3.8 まとめ	39

エグゼクティブサマリ

企業ユーザのクラウド利用は年々増加しています。総務省の平成26年度版通信白書によると、クラウドサービスを利用している企業の割合は、平成25年末には33.1%となり、24年末の28.2%からおよそ5%増加しています。これは約3分の1の企業が何らかのクラウドサービスを利用しており、クラウドもいよいよ普及期に入ったことを示しています。

一方、同白書によるとクラウドサービスを利用していない企業のうち37.4%がセキュリティ面の不安を理由に挙げています。8月に起こった大勢の有名芸能人のプライベート写真が流出した事件では、ユーザがスマートフォンで撮影し、クラウドに自動転送されて保管された写真が、そのユーザのクラウドアカウントへの不正ログインを通じて流出しました。何らかのシステムの脆弱性を突かれたものではなく、不正ログインによるものですので、これは流出被害を受けた人の自己責任ということになります。同白書によると日本のスマートフォンの保有率は53.5%に至っており、これを業務に活用する機会も増えると考えられますが、上記の例のような機密情報の流出を防ぐためには、ユーザー一人一人が、自分が扱うデータのセキュリティ確保に必要な最新知識を持ち、自ら安全を確保するための適切な対策を取ることがますます重要になるでしょう。

本レポートは、このような状況の中で、IJがインターネットというインフラを支え、お客様に安心・安全に利用し続けていただくために継続的に取り組んでいる様々な調査・解析の結果や、技術開発の成果、ならびに、重要な技術情報を定期的にとりまとめ、ご提供するものです。

「インフラストラクチャセキュリティ」の章では、2014年7月から9月までの3ヵ月間に発生した主なインシデントを時系列に並べ、分類し、月ごとに概要をまとめると共に、期間全体での統計と解析結果をご報告します。また、対象期間中のフォーカスリサーチとして、9月に公開された"ShellShock"と呼ばれるBashの脆弱性と、10月に公開された"POODLE attack"と呼ばれるSSLv3に対する新たな攻撃手法、及び、リスト型攻撃と呼ばれるアカウントハッキング手法について、それぞれの概要と対策について解説します。

「コンテンツ配信」の章では、近年増加の一途を辿り、2018年には全世界のコンシューマトラフィックの約80～90%になると予測されているビデオストリーミングの最新の状況について説明します。特に、4Kのような超高画質画像のビデオストリーミングを可能とするCODEC規格H.265と、インターネット上での安定かつ効率的な配信を実現するMPEG-DASHの特徴を解説します。またIJが今年実施した、夏の甲子園のライブ配信の実例を紹介します。

「クラウドコンピューティングテクノロジー」の章では、ネットワークを仮想化し、ソフトウェアにより制御可能とするためのSDNの概念とOpenFlowプロトコルについての基本をおさらいし、総務省のSDN普及に向けた戦略的研究開発プロジェクト「O3」の成果として今年オープンソース化された、OpenFlow対応ソフトウェアスイッチ「Lagopus」を紹介します。特に、従来のソフトウェアスイッチと比較して飛躍的な性能向上を実現し、10GbEワイヤードでのパケット転送を達成した実装技術について解説します。

IJでは、このような活動を通じて、インターネットの安定性を維持しながらも、日々改善し発展させて行く努力を続けております。今後も、お客様の企業活動のインフラとして最大限に活用していただくべく、様々なソリューションを提供し続けて参ります。

執筆者:



浅羽 登志也(あさば としや)

株式会社IJイノベーションインスティテュート 代表取締役社長。株式会社ストラトスフィア 代表取締役社長。1992年、IJの設立と共に入社し、バックボーンの構築、経路制御、国内外ISPとの相互接続などに従事。1999年より取締役、2004年より取締役副社長として技術開発部門を統括。2008年6月に株式会社IJイノベーションインスティテュートを設立、同代表取締役社長に就任。2012年4月に株式会社ストラトスフィアを設立、同代表取締役社長に就任。

Bashの脆弱性Shellshockについて

今回は、話題となったBashの脆弱性Shellshockについて、SSLv3への新たな攻撃手法POODLE attackについて、昨年発生し続けているリスト型攻撃の発生状況と対策について解説します。

1.1 はじめに

このレポートは、インターネットの安定運用のためにIJが取得した一般情報、インシデントの観測情報、サービスに関連する情報、協力関係にある企業や団体から得た情報を元に、IJが対応したインシデントについてまとめたものです。今回のレポートで対象とする2014年7月から9月までの期間では、国内の報道機関など複数のサイトにおいて、DNSハイジャックによりマルウェア感染に誘導される事件が発生しました。また、非常に多くの製品やサービスに影響するGNU Bashの脆弱性が発見されています。オンラインサービスに対するリスト型攻撃による不正ログイン事件や、オンラインバンキングを悪用した不正送金事件が依然として継続的に発生しています。DDoS攻撃においては、DNSやNTPに続いてSSDPを踏み台とする攻撃が国内においても発生しました。国外では、200Gbpsを超えるようなDDoS攻撃が散発的に発生しています。このように、インターネットでは依然として多くのインシデントが発生する状況が続いています。

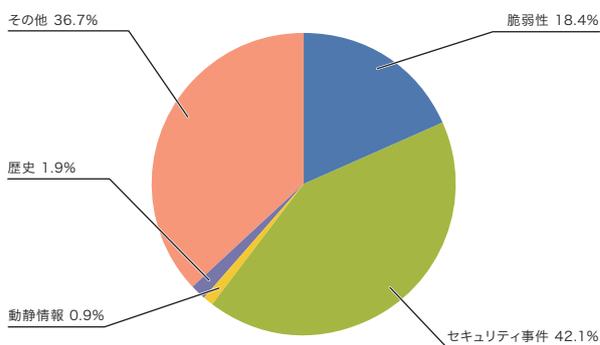


図-1 カテゴリ別比率(2014年7月~9月)

1.2 インシデントサマリ

ここでは、2014年7月から9月までの期間にIJが取り扱ったインシデントと、その対応を示します。まず、この期間に取り扱ったインシデントの分布を図-1に示します*1。

■ Anonymousなどの活動

この期間においても、Anonymousに代表されるHacktivistによる攻撃活動は継続しています。様々な事件や主張に応じて、多数の国の企業や政府関連サイトに対するDDoS攻撃や情報漏えい事件が発生しました。7月から8月にかけて、パレスチナ自治区ガザでの紛争に関連し、イスラエルの複数の政府関連サイトや民間企業のWebサイトに対し、Web改ざんやDDoS攻撃、情報漏えいの被害が発生しています(OpSaveGaza)。紛争に関連するところでは、ロシアとウクライナ、インドとインドネシア、インドとパキスタンなどで、相互に攻撃が行われています。8月から9月にかけては、パキスタン政府への抗議から、複数の政府関連サイトに対するWeb改ざんやDDoS攻撃、SQLインジェクションによる情報漏えいの被害が発生しました(OpPakistan)。9月には、香港での選挙制度の問題に対するデモ活動と関連して中国政府や香港行政政府のWebサイトに対するDDoS攻撃やメールアドレスの漏えいなどが発生しています(OpHongKong)。

他にも、世界中の各国政府とその関連サイトに対して、AnonymousなどのHacktivist達による攻撃が継続して行われました。また、Syrian Electronic Armyを名乗る何者かによるSNSアカウントの乗っ取りやWebサイト改ざんも継続して発生しており、被害を受けたアカウントにはイスラエル国防軍なども含まれていました。

*1 このレポートでは、取り扱ったインシデントを、脆弱性、動静情報、歴史、セキュリティ事件、その他の5種類に分類している。
脆弱性: インターネットや利用者の環境でよく利用されているネットワーク機器やサーバ機器、ソフトウェアなどの脆弱性への対応を示す。
動静情報: 要人による国際会議や、国際紛争に起因する攻撃など、国内外の情勢や国際的なイベントに関連するインシデントへの対応を示す。
歴史: 歴史上の記念日などで、過去に史実に関連して攻撃が発生した日における注意・警戒、インシデントの検知、対策などの作業を示す。
セキュリティ事件: ワームなどのマルウェアの活性化や、特定サイトへのDDoS攻撃など、突発的に発生したインシデントとその対応を示す。
その他: イベントによるトラフィック集中など、直接セキュリティに関わるものではないインシデントや、セキュリティ関係情報を示す。

■ 脆弱性とその対応

この期間中では、Microsoft社のWindows^{*2*3}、Internet Explorer^{*4*5*6}などで修正が行われました。Adobe社のAdobe Flash Player、Adobe Reader及びAcrobatでも修正が行われました。Oracle社のJava SEでも四半期ごとに行われている更新が提供され、多くの脆弱性が修正されました。これらの脆弱性のいくつかは、修正が行われる前に悪用が確認されています。サーバアプリケーションでは、データベースサーバとして利用されているOracleを含むOracle社の複数の製品で四半期ごとに行われている更新が提供され、多くの脆弱性が修正されました。

Cisco Unified Communications Domain Managerには、権限昇格を含む複数の脆弱性が見つかри、修正されています。修正された脆弱性には初期設定のSSH秘密鍵の利用による脆弱性が含まれていました。機器の初期設定のSSH秘密鍵を更新せず利用することは、同じ秘密鍵を持つ第三者から悪用される可能性が高いことから行うべきではありません^{*7}。

LinuxなどUNIX系のOSで利用されているシェルのBashでは、任意のOSコマンドが実行できる脆弱性(CVE-2014-6271)が見つかリ、修正されました。この脆弱性はShellshockと呼ばれ、CGIプログラムが動くWebサーバ、ルータやゲートウェイ製品など、多くのアプリケーションや機器に影響があったことから問題となりました。この問題の詳細については「1.4.1 Bashの脆弱性Shellshockについて」を併せてご参照ください。

■ なりすましによる不正ログイン

この期間でも、昨年から多数発生しているユーザのIDとパスワードを狙った試みと、取得したIDとパスワードのリストを使用したと考えられる不正ログインの試みが継続して発生しています。アンケートサイト、インターネット通販サイト、物流事業者のサポートサイト、携帯電話会社のサイト、SNSなど様々なサイトが攻撃対象となっています。このうちのいくつかの事件では、サイト上のポイントを他サービスのギフトポイントに不正に交換されるなどの被害も発生しています。このように、リスト型攻撃の脅威が依然として続いていることから、様々な企業のWebサービスやアプリケーションで、2段階認証の導入やパスワードの文字数や使用できる文字の制限緩和などの認証機能の強化といった対策が進められています。また、ユーザが安易な使い回しを行ったり、推測されやすい簡易なパスワードを設定するなどの問題も指摘されていることから^{*8}、利用者側も安全な利用をする上で注意が必要です。詳細については「1.4.3 リスト型攻撃の発生状況と対策」も併せてご参照ください。

■ Web改ざんと不正なソフトウェアへの誘導

この期間では、前回の期間に続いてWebサイトの改ざんによる不正なソフトウェアへ誘導される事件が多く発生しました。ツアー会社、自動車販売会社、セキュリティ企業のWebサイトでの改ざんとそれによる不正なソフトウェアへの誘導が発生しています。また、不正なソフトウェアへの誘導だけでなく、独立行政法人のWebサイトが改ざんされた事件では、Web改ざんによってフィッシングサイトとして悪用しようとする試みも発生しています^{*9}。また、テキスト

*2 「マイクロソフト セキュリティ情報 MS14-038 - 緊急 Windows Journalの脆弱性により、リモートでコードが実行される(2975689)」(<https://technet.microsoft.com/ja-JP/library/security/ms14-038.aspx>)。

*3 「マイクロソフト セキュリティ情報 MS14-043 - 緊急 Windows Media Centerの脆弱性により、リモートでコードが実行される(2978742)」(<https://technet.microsoft.com/ja-JP/library/security/ms14-043.aspx>)。

*4 「マイクロソフト セキュリティ情報 MS14-037 - 緊急 Internet Explorer用の累積的なセキュリティ更新プログラム(2975687)」(<https://technet.microsoft.com/ja-jp/library/security/ms14-037.aspx>)。

*5 「マイクロソフト セキュリティ情報 MS14-051 - 緊急 Internet Explorer用の累積的なセキュリティ更新プログラム(2976627)」(<https://technet.microsoft.com/ja-JP/library/security/ms14-051.aspx>)。

*6 「マイクロソフト セキュリティ情報 MS14-052 - 緊急 Internet Explorer用の累積的なセキュリティ更新プログラム(2977629)」(<https://technet.microsoft.com/ja-JP/library/security/ms14-052.aspx>)。

*7 同じ秘密鍵を利用するリスクについては、IJ Security Diary、「SSL/TLS、SSHで利用されている公開鍵の多くが意図せず他のサイトと秘密鍵を共有している問題」(<https://sect.ij.ad.jp/d/2012/08/109998.html>)も参照のこと。

*8 例えば、IPAが実施した「『オンライン本人認証方式の実態調査』報告書について」(<http://www.ipa.go.jp/security/fy26/reports/minsho/index.html>)などを参照のこと。

*9 例えば、次の独立行政法人 防災科学技術研究所の発表を参照のこと。「防災科研公開 web に対する改ざんについて」(http://www.bosai.go.jp/press/2014/pdf/20140811_01.pdf)。

7月のインシデント

1	脆弱	3日: Cisco Unified Communications Domain Managerに権限昇格を含む複数の脆弱性が見つかり、修正された。 "Multiple Vulnerabilities in Cisco Unified Communications Domain Manager" (http://www.cisco.com/cisco/web/support/JP/112/1122/1122753_cisco-sa-20140702-cucdm-j.html)。
2		
3	脆弱	9日: Microsoft社は、2014年7月のセキュリティ情報を公開し、MS14-037とMS14-038の2件の緊急と3件の重要な更新を含む合計6件の修正をリリースした。 「2014年7月のマイクロソフト セキュリティ情報の概要」(https://technet.microsoft.com/ja-JP/library/security/ms14-jul)。
4		
5	脆弱	9日: Adobe Flash Playerに、任意のコード実行の可能性がある複数の脆弱性が発見され、修正された。 「APSB14-17: Adobe Flash Player用のセキュリティアップデート公開」(http://helpx.adobe.com/jp/security/products/flash-player/apsb14-17.html)。
6	セキュリティ	9日: 通信教育企業は、同社の約2070万人分の顧客情報が名簿業者など外部に流出したことを公表した。その後、9月に公表された最終報告では約4858万人分の個人情報が入り混じったことが判明している。
7	セキュリティ	9日: インド政府のルート認証局の傘下で中間認証局を運営するインド国立情報工学センター(NIC)で複数のGoogleドメインやYahoo!ドメインの証明書が不正に発行されたことが判明し、複数のブラウザで当該証明書を無効にする対応が行われた。原因については証明書発行プロセスが破られたためとされている。 Google Online Security Blog, "Maintaining digital certificate security" (http://googleonlinesecurity.blogspot.jp/2014/07/maintaining-digital-certificate-security.html)。
8		
9		
10	他	15日: 6月に成立した改正児童買春・ポルノ禁止法が施行された。ただし、自己の性的好奇心を満たす目的での児童ポルノの所持等を処罰する改正法7条1項の規定については、適切に廃棄などの措置ができるよう猶予期間として、施行の日から1年間は適用しないことが附則に定められたことから平成27年7月15日から適用される。 詳細については次の法務省による解説も参照のこと。「児童買春、児童ポルノに係る行為等の処罰及び児童の保護等に関する法律の一部を改正する法律案」(http://www.moj.go.jp/keiji1/keiji1_1_00008.html)。
11		
12	他	15日: 総務省より、情報通信の現況及び情報通信の政策の動向について掲載した「情報通信白書平成26年版」が公表された。 情報通信白書ホームページ(http://www.soumu.go.jp/johotsusintokei/whitepaper/index.html)。
13		
14	脆弱	16日: Oracle社は、Oracleを含む複数製品について、四半期ごとの定例アップデートを公開し、Java SEの20件の脆弱性を含む合計113件の脆弱性を修正した。 "Oracle Critical Patch Update Advisory - July 2014" (http://www.oracle.com/technetwork/jp/topics/ojkbcpujul2014-2244696-ja.html)。
15	他	16日: IPAは、標的型攻撃を受けた組織に対する被害拡大と再発の抑止・低減、速やかな対策による連鎖の遮断などの支援を行うサイバーレスキュー隊を正式に発足した。 「プレス発表 標的型サイバー攻撃への対策支援『サイバーレスキュー隊』を発足」(https://www.ipa.go.jp/about/press/20140716_1.html)。
16		
17	セキュリティ	17日: 通信教育企業の顧客情報が外部に漏えいした事件について、業務委託企業の元派遣社員が不正競争防止法違反(営業秘密の複製)の容疑で逮捕された。
18		
19	セキュリティ	18日: 一般財団法人日本データ通信協会テレコム・アイザック推進会議は、インターネットバンキングに係るマルウェア(Game Over Zeus)の国際的な感染駆除作戦に関連し、官民連携による国民のマルウェア対策支援プロジェクト(ACTIVE)を通じて、該当マルウェアに感染している利用者への注意喚起を実施することを発表した。 「インターネットバンキングに係るマルウェア感染者に対する注意喚起について」(https://www.telecom-isac.jp/news/news20140718.html)。
20		
21		
22	他	22日: インターネットの安定的な運用に関する協議会より、「電気通信事業者における大量通信等への対処と通信の秘密に関するガイドライン」の第3版が公開された。 「電気通信事業者における大量通信等への対処と通信の秘密に関するガイドライン(第3版)」(http://www.jaipa.or.jp/other/mtcs/guideline_v3.pdf)。
23		
24	他	23日: 総務省は、行政機関及び独立行政法人などが保有するパーソナルデータについて、その特質を踏まえた調査・検討を行うため、「行政機関等が保有するパーソナルデータに関する研究会」を開催することを発表した。 「行政機関等が保有するパーソナルデータに関する研究会」(http://www.soumu.go.jp/main_sosiki/kenkyu/gyousei_personal/index.html)。
25		
26	セキュリティ	24日: 欧州中央銀行は、Webサイトに侵入され、イベント登録者の電子メールアドレスなどの個人データが漏えいしたことを公表した。この事件は漏えいしたデータと引き換えに金銭を要求する匿名のメールが届いたことから発覚した。 詳細については次の欧州中央銀行の発表を参照のこと。"24 July 2014 - ECB announces theft of contact information" (https://www.ecb.europa.eu/press/pr/date/2014/html/pr140724.en.html)。
27		
28		
29	セキュリティ	25日: 国内の複数の企業や行政機関より、自サイトのホームページを模倣したWebサイトに対する注意喚起が相次いで行われた。
30	脆弱	29日: 複数のIPカメラに認証回避の脆弱性があり、認証情報を含む機器の設定内容が取得された上で任意の操作が実行される可能性があり、修正された。 JVN、「JVNDB-2014-000087 アイ・オー・データ機器製の複数のIPカメラにおける認証回避の脆弱性」(http://jvndb.jvn.jp/ja/contents/2014/JVNDB-2014-000087.html)。
31		

[凡例] 脆弱 脆弱性 | セキュリティ事件 | 動 動静情報 | 歴 歴史 | 他 その他

※日付は日本標準時

編集ソフトのサポートサイトで発生したWeb改ざん事件では、アクセスしたユーザが別サイトに誘導されて不正なソフトウェアのインストールが行われただけでなく、事件発覚後に正規のアップデートファイルのコンテンツが改ざんされ、ウイルスを含んだファイルを利用者にアップデートファイルとしてインストールさせようとしていたことも判明しています。このような、正規ソフトウェアのアップデートの仕組みを悪用したマルウェアの感染活動は今後も継続すると考えられます。

■ 動静や歴史的背景による攻撃

この期間では毎年、太平洋戦争の歴史的日付や、竹島や尖閣諸島などに関連したインシデントが発生しています。本年もこれらに関連した日本国内の複数の政府機関や民間企業のWebサイトに対し、SQLインジェクションや、ブルートフォースなどによる侵入による改ざんやDDoS攻撃が発生すると予測されたため、警戒を行いました。しかし、一部でSQLインジェクション攻撃などの不正なアクセスが増加していたり^{*10}、Webサイトの改ざんが確認されていますが^{*11}、大規模な攻撃の発生は確認されませんでした。IJの観測では、DDoS攻撃が平常時よりも若干多く見られましたが、攻撃の規模や回数は過去の同期間に比べるとかなり減少しています。

■ DDoS攻撃

この期間では、大規模なDDoS攻撃がいくつか発生しています。8月にはLizard Squadを名乗る何者かによるPSNやXbox LIVE、League of Legendsなど複数のゲームサーバに対する攻撃が発生しています。PSNの事件では263.35GbpsのNTPリフレクション攻撃を受けたとされています。更に、この事件ではDDoS攻撃だけでなく、役員が搭乗していた航空機に爆発物を仕掛けたなどの発言を犯人がオンライン上で行ったことから、実際に搭乗していた航空機の運航に支障が出るなどの影響がありました。この攻撃者によると思われるゲーム関連サーバに対する攻撃は、9月に入っても断続的に発生しています。日本では、5月末に複数のISPにおいて、

DNSサーバに対する断続的なDDoS攻撃が発生していますが、一部のISPに対しては7月に入っても継続しています。また、3月に海外のDDoS攻撃の代行サービスを利用して、ゲームサーバに対する攻撃を行ったとして、9月に電子計算機損壊等業務妨害容疑で高校生が書類送検されました。

■ 企業におけるマルウェア感染と情報漏えい

この期間、企業のシステムがマルウェア感染したことによる大規模な顧客情報などの漏えい事件が引き続き発生しています。特に米国では、8月に物流大手企業で、国内の51カ所の代理店でマルウェア感染が確認され、顧客のクレジットカード情報などが流出した可能性があることが発表されています。日本でも航空会社で社内の端末へのマルウェア感染によって、顧客情報が外部に送信され漏えいした可能性のある事件などが発生しています。9月には、ホームセンター大手企業で、約5600万枚分の決済カード情報や電子メールアドレスが流出した可能性のある事件が発生しました。この事件では昨年から発生しているPOSシステムを狙ったマルウェアの亜種が使われたとされており、クレジットカードなどの情報が漏えいした可能性がありました。これ以外にも病院や小売り業者など複数の企業でマルウェアによる情報漏えい事件が発生しています。特に、POSマルウェアについては、昨年終わりがより大規模な情報漏えい事件を複数回引き起こしており、US-CERTからも複数回注意喚起が行われています^{*12}。新種のPOSマルウェアが継続して確認されていることから^{*13}、今後も引き続き注意が必要です。

■ 政府機関の取り組み

政府機関のセキュリティ対策の動きとしては、政府の情報セキュリティ政策会議第40回会合が開催され、2013年度の我が国におけるサイバーセキュリティ全般の状況について、政府機関、重要インフラ事業者、各府省庁の関連施策の実施状況、関係資料などを1つに取りまとめた初めての年次報告書である「サイバーセキュリティ政策に係る年次報告(2013年度)」や、各府省庁のサイバーセキュリティに

*10 例えば、IBM社のTokyo SOC Reportでは特定の国からのSQLインジェクションなどの攻撃が増加傾向であることが報告されている。「柳条湖事件が起こった9月18日前後の攻撃動向について」(<https://www-304.ibm.com/connections/blogs/tokyo-soc/?lang=ja>)。

*11 Web改ざんについては次のエッセイブログなどを参照のこと。「9.18のサイバー攻撃に関して(追記)」(<http://blog.f-secure.jp/archives/50734688.html>)。

*12 US-CERTでは、1月に"TA14-002A: Malware Targeting Point of Sale Systems" (www.us-cert.gov/ncas/alerts/TA14-002A)、8月に"Alert (TA14-212A) Backoff Point-of-Sale Malware" (<https://www.us-cert.gov/ncas/alerts/TA14-212A>)と、継続的に注意喚起を行っている。

*13 例えば、トレンドマイクロ社のブログ「急増するPOSシステムへの攻撃とPOSマルウェアファミリ」(<http://blog.trendmicro.co.jp/archives/9902>)などを参照のこと。

8月のインシデント

1	セ	1日:US-CERTは、POSシステムを対象とした新種の不正プログラムBackoffが確認されたとして、注意喚起を行った。 "Alert (TA14-212A) Backoff Point-of-Sale Malware"(https://www.us-cert.gov/ncas/alerts/TA14-212A)。
2	他	1日:Microsoft社は、アプリケーションの脆弱性を緩和するセキュリティツールであるEnhanced Mitigation Experience Toolkit(EMET)5.0を公開した。 詳細については次のTechNet Blogsなどを参照のこと。「EMET 5.0 公開しました」(http://blogs.technet.com/b/jpsecurity/archive/2014/08/01/emet-5-released.aspx)。
3		
4	セ	2日:Mozilla Developer Networkは、データベースダンプファイルが誤って公開状態になっていたことから、7万6千のMDNユーザのメールアドレスと4,000ユーザの暗号化されたパスワードが漏えいした可能性があることを公表した。 Mozilla Developer Network、「MDNデータベースの情報漏洩について」(http://www.mozilla.jp/blog/entry/10418/)。
5		
6		
7	セ	4日:各国の政府機関が情報収集活動に利用しているとされる商用監視ソフトウェアFinSpy(FinFisher)について、提供元のGamma Internationalが不正アクセスを受け、40GBにもなる内部文書とソースコードが公開された。
8		
9	セ	8日:米国のセキュリティ企業より、2014年2月から5月にかけて、BGPハイジャックにより、仮想通貨のマイニングプールへのトラフィックを偽のマイニングプールに振り向ける攻撃が行われていたことが報告された。この攻撃では、19のISPが影響を受けたとされており、攻撃者は8万3000ドルの利益を得ていた可能性があることが指摘されている。 詳細については次のDell SecureWorksのブログを参照のこと。「BGP Hijacking for Cryptocurrency Profit」(http://www.secureworks.com/cyber-threat-intelligence/threats/bgp-hijacking-for-cryptocurrency-profit/)。
10		
11		
12		
13	セ	12日:米国で、インターネットの速度が低下したり、通信が不安定になるなどの現象が発生した。原因については、広報されたBGPの経路情報が、古いルーターでBGPのルーティングテーブルの上限である512kを超えたためと考えられる。 詳細については例えば、次のBGPmon.netのブログを参照のこと。「What caused today's Internet hiccup」(http://www.bgpmon.net/what-caused-todays-internet-hiccup/)。
14		
15		
16	脆	13日:Microsoft社は、2014年8月のセキュリティ情報を公開し、MS14-043とMS14-051の2件の緊急と7件の重要な更新をリリースした。 「2014年8月のマイクロソフト セキュリティ情報の概要」(https://technet.microsoft.com/ja-JP/library/security/ms14-aug)。
17	脆	13日:Adobe Flash Playerに、任意のコード実行の可能性がある複数の脆弱性が発見され、修正された。 「APSB14-18: Adobe Flash Player用のセキュリティアップデート公開」(http://helpx.adobe.com/jp/security/products/flash-player/apsb14-18.html)。
18	脆	13日:Adobe Reader及びAcrobatに、リモートから任意のコード実行の可能性がある脆弱性が発見され、修正された。 「APSB14-19: Adobe ReaderおよびAcrobat用セキュリティアップデート公開」(http://helpx.adobe.com/jp/security/products/reader/apsb14-19.html)。
19	セ	13日:テキスト編集ソフトの日本語と簡体字中国語のサポートサイトが改ざんされ、利用者のユーザ名、パスワード、IPアドレスを盗もうとする痕跡が見つかっことが公表された。その後、8月18日に再度Webサイトが改ざんされ、当該シェアウェアの更新チェック機能を利用して悪意あるファイルがインストールされる事件も発生している。
20		
21		
22	脆	15日:Microsoft社は、2014年8月に修正をリリースしたMS14-045について、適用した場合には異常終了や起動失敗が発生する可能性があるとして、問題が発生していない場合でも、予防的処置としてこれらの更新プログラムをアンインストールすることを推奨すると発表した。 詳細については、次の日本のセキュリティチーム公式ブログによる解説も参照のこと「【リリース後に確認された問題】2014年8月13日公開の更新プログラムの適用により問題が発生する可能性がある」(http://blogs.technet.com/b/jpsecurity/archive/2014/08/16/2982791-knownissue3.aspx)。
23		
24		
25	セ	21日:米国UPSは、国内の51カ所の代理店でマルウェア感染が確認され、顧客のクレジットカード情報などが流出した可能性があることを公表した。 詳細については次のUnited Parcel Service of America, Inc.の発表などを参照のこと。「The UPS Store, Inc. Notifies Customers Of Potential Data Compromise and Incident Resolution」(http://www.pressroom.ups.com/Press+Releases/Archive/2014/Q3/The+UPS+Store%2C+Inc.+Notifies+Customers+Of+Potential+Data+Compromise+and+Incident+Resolution)。
26		
27		
28	セ	25日:何者かによって、PlayStation Network (PSN)及びSony Entertainment Network (SEN)へのDDOS攻撃が行われ、大規模な障害が発生した。 攻撃については、例えば次のPlayStation.Blogなどを参照のこと。「Update: PlayStation Network is Back Online」(http://blog.us.playstation.com/2014/08/24/playstation-network-update-2/)。
29		
30		
31	脆	28日:Microsoft社は、適用後に異常終了や起動失敗などの不具合が発生していたMS14-045について、修正を行い、再公開した。 「マイクロソフト セキュリティ情報 MS14-045 重要 カーネルモード ドライバーの脆弱性により、特権が昇格される (2984615)」(https://technet.microsoft.com/ja-jp/library/security/ms14-045.aspx)。

[凡例] 脆 脆弱性 セ セキュリティ事件 動 動 静 情 報 歴 歴 史 他 その他

※日付は日本標準時

関する今年度の施策などを取りまとめた、「サイバーセキュリティ 2014」などが決定しています*14。

本年6月に公表された「パーソナルデータの利活用に関する制度改正大綱」を受け、行政機関及び独立行政法人などが保有するパーソナルデータについて、その特質を踏まえた調査・検討を行うため、「行政機関等が保有するパーソナルデータに関する研究会」が総務省にて開催されています。同じく総務省より、電気通信事業者が取り扱う位置情報について、通信の秘密や個人情報・プライバシーを適切に保護しながら、ビジネス利用も含めたその社会的利活用を促進するため、位置情報の取得、利用及び第三者提供時における適切な取り扱いについて取りまとめた「位置情報プライバシーレポート～位置情報に関するプライバシーの適切な保護と社会的利活用の両立に向けて～」も公表されています*15。

警察庁からは、平成26年度の警察白書が発表され、サイバー犯罪の検挙件数が過去最多を記録したことや、インターネットバンキングに係る不正送金事犯が急増していることなどが記載されています。また、サイバー空間の脅威に対する官民の連携の推進を進めているなどの取り組みについても解説されています。

■ その他

6月より対策活動が実施されている、オンラインバンキングなどの情報窃取を行うマルウェアであるGameOver Zeusについて*16、7月には本作戦で得られた該当マルウェアの感染端末に関する情報を元に、ISP事業者に対し、感染者に関する情報提供とそれを用いた注意喚起が実施され

ることが発表されています*17。また、このような国際的なサイバー犯罪に対処するため、9月には欧州刑事警察機構(Europol)が、欧州や米国、カナダなどを含む複数国によるJoint Cybercrime Action Taskforce(J-CAT)を発足させるなど*18、国際的な連携の枠組みの構築も進められています。

7月には、様々な団体や企業から模倣したWebサイトについての注意喚起が行われました*19。8月に入ってから、政府機関でも同様の事例について注意喚起が行われています。これらの事例は特定の企業や団体などを狙ったものではなく、Webプロキシサービスによるものと考えられます。同様の事例としては、昨年中国の携帯電話事業者が提供していたWebページの変換サービス用プロキシによる事例があります*20。他にも同様のサービスを提供している場合が多くありますが、提供者がはっきりとしていないことも多いため、利用には注意が必要です*21。

同じく7月には、通信教育企業の顧客情報が漏えいした事件が明るみに出ました。これは別の企業が入手した名簿を元にダイレクトメールを送っていたことで判明したものです。情報漏えいした企業の業務委託企業の元派遣社員が不正に持ち出して名簿業者などに販売していたとして、不正競争防止法違反(営業秘密の複製)罪で逮捕されています。

電気通信事業者がDoS攻撃などの大量通信を識別し、その対処を適法に実施するためのガイドライン「電気通信事業者における大量通信等への対処と通信の秘密に関するガイドライン」が改定され、電気通信事業関連5団体より公表されています*22。これは、総務省で行われていた、電気通信事

*14 内閣官房情報セキュリティセンター、「情報セキュリティ政策会議 第40回会合(平成26年7月10日)」(<http://www.nisc.go.jp/conference/seisaku/index.html#seisaku40>)。

*15 総務省、「緊急時等における位置情報の取扱いに関する検討会 報告書『位置情報プライバシーレポート～位置情報に関するプライバシーの適切な保護と社会的利活用の両立に向けて～』の公表」(http://www.soumu.go.jp/menu_news/s-news/01kiban08_02000144.html)。

*16 Department of Justice, "U.S. Leads Multi-National Action Against "Gameover Zeus" Botnet and "Cryptolocker" Ransomware, Charges Botnet Administrator" (<http://www.justice.gov/opa/pr/2014/June/14-crm-584.html>)。

*17 一般財団法人日本データ通信協会テレコム・アイザック推進会議、「インターネットバンキングに係るマルウェア感染者に対する注意喚起について」(<https://www.telecom-isac.jp/news/news20140718.html>)。

*18 詳細については次のEuropolの発表を参照のこと。"EXPERT INTERNATIONAL CYBERCRIME TASKFORCE IS LAUNCHED TO TACKLE ONLINE CRIME" (<https://www.europol.europa.eu/content/expert-international-cybercrime-taskforce-launched-tackle-online-crime>)。

*19 例えば、次の大阪府警察などの発表を参照のこと「緊急:大阪府警察ホームページを模倣したウェブサイトにご注意!」(http://www.police.pref.osaka.jp/15topics/caution_domain.html)。

*20 例えば、次のJPCERTコーディネーションセンターのつぶやき(<https://twitter.com/jpcert/status/322282948554530816>)を参照のこと。

*21 この事例については次のトレンドマイクロ社のブログ『プロキシ回避システム』がもたらした『模倣サイト』の混乱、その事例から学ぶ教訓とは?」(<http://blog.trendmicro.co.jp/archives/9713>)なども参照のこと。

*22 インターネットの安定的な運用に関する協議会は電気通信事業関連の5つの業界団体により構成されている。このガイドラインの策定については次を参照のこと。社団法人日本インターネットプロバイダー協会(JAIPA)「電気通信事業者における大量通信等への対処と通信の秘密に関するガイドラインの改定について」(<http://www.jaipa.or.jp/topics/?p=695>)。

9月のインシデント

1	セ	1日：米国で、ハリウッド女優などのプライベート写真が、掲示板に掲載される事件が発生した。この事件は、流出した複数の著名人のiCloudアカウントに不正アクセスが行われたことが原因とされている。
2		調査を実施したApple社は次の発表を行っている。"Apple Media Advisory Update to Celebrity Photo Investigation"(http://www.apple.com/pr/library/2014/09/02Apple-Media-Advisory.html)。
3	他	1日：総務省より、平成25年度に発生した電気通信事故の報告状況を取りまとめた「電気通信サービスの事故発生状況(平成25年度)」が公表された。
4		「電気通信サービスの事故発生状況(平成25年度)」(http://www.soumu.go.jp/menu_news/s-news/01kiban05_02000072.html)。
5	セ	3日：米国の小売り大手であるHome Depotで大規模なカード情報の流出が発生した。
6		詳細については、次のThe Home Depot社の発表を参照のこと。"The Home Depot Reports Findings in Payment Data Breach Investigation"(https://corporate.homedepot.com/MediaCenter/Documents/Press%20Release.pdf)。
7		
8	他	4日：警察庁は、平成26年上半年期のインターネットバンキングに係る不正送金事犯の発生状況を公表した。被害金額が平成25年下半年期に続き、増加していることや、対象が地方銀行や信用金庫、信用組合に拡大していること、法人名義の口座に対する被害が急増していることなどが挙げられている。
9		「平成26年上半年期のインターネットバンキングに係る不正送金事犯の発生状況について」(http://www.npa.go.jp/cyber/pdf/H260904_banking.pdf)。
10		
11	脆	10日：Microsoft社は、2014年9月のセキュリティ情報を公開し、MS14-052の緊急と3件の重要な更新をリリースした。
12		「2014年9月のマイクロソフト セキュリティ情報の概要」(https://technet.microsoft.com/ja-JP/library/security/ms14-sep)。
12	脆	10日：Adobe Flash Playerに、不正終了や、任意のコード実行の可能性がある複数の脆弱性が発見され、修正された。
13		「APSB14-21: Adobe Flash Player用のセキュリティアップデート公開」(http://helpx.adobe.com/jp/security/products/flash-player/apsb14-21.html)。
14	他	11日：警察庁は、平成26年上半年期のサイバー犯罪の傾向などをまとめた「平成26年上半年期のサイバー空間をめぐる脅威の情勢について」を公表した。
15		「平成26年上半年期のサイバー空間をめぐる脅威の情勢について」(http://www.npa.go.jp/kanbou/cybersecurity/H26_kami_jousei.pdf)。
16		
17	脆	17日：Adobe Reader及びAcrobatに、不正終了や、リモートから任意のコード実行の可能性がある脆弱性が発見され、修正された。
18		「APSB14-20: Adobe ReaderおよびAcrobat用セキュリティアップデート公開」(http://helpx.adobe.com/jp/security/products/reader/apsb14-20.html)。
19	セ	18日：ゲームサーバにDDoS攻撃を複数回払い、ゲーム会社の業務を妨害したとして高校生が電子計算機損壊等業務妨害容疑で書類送検された。
20	動	18日：毎年、歴史的な要因によりこの日の前後に発生していた攻撃については、小規模な攻撃はあったが組織的な攻撃は見られなかった。
21	他	22日：LINE株式会社は、多発しているLINEアカウントの乗っ取り被害を抑止する対策の1つとして、スマートフォン版LINEアプリでPINコード(4桁の番号)の設定を必須とする措置を講じた。
22		詳細については次のLINE公式ブログ「【重要】不正ログイン(乗っ取り)の被害拡大を防ぐため、PINコードの設定を必須にします」(http://official-blog.line.me/ja/archives/1009539887.html)を参照のこと。
23		
24	セ	24日：航空会社でマルウェア感染による不正アクセスが発生し、最大で73万件の会員の個人情報が漏えいした可能性があることが公表された。
25		
26	脆	25日：Bashにリモートから任意のコード実行が可能な脆弱性が見つかり、修正された。その後、修正が不十分なことが判明したことから、複数の脆弱性が改めて修正されている。
27		JVN、「JVN#97219505 GNU BashにOSコマンドインジェクションの脆弱性」(http://jvn.jp/vu/JVNVU97219505/)。
28		
29	脆	26日：大手クラウドプロバイダが未公開のXenの脆弱性を理由に大規模なメンテナンスの実施を行い話題となった。10月2日になり、CVE-2014-7188への対応だったことが公表された。
30		詳細については次のXen Project Blogなどを参照のこと。"XSA-108: Additional Information from the Xen Project"(https://blog.xenproject.org/2014/10/02/xsa-108-additional-information-from-the-xen-project-2/)。
30	セ	26日：物流事業者のサポートサイトへパスワードリスト攻撃と考えられる不正ログインがあり、一部の会員の個人情報が閲覧されることで漏えいした可能性があることが公表された。
		同様の事件は28日に別の物流事業者のサポートサイトでも発生している。

[凡例] 脆 脆弱性 セ セキュリティ事件 動 動静情報 歴 歴史 他 その他

※日付は日本標準時

業におけるサイバー攻撃への適正な対処の在り方に関する研究会の「第一次とりまとめ」が4月に公表されたこと^{*23}を踏まえて追加修正を行ったもので、DNSアンプ攻撃などの新たな脅威への対応が図られています。

また、9月初旬から10月にかけて、国内の複数の組織で利用していた.comドメインがDNSハイジャックを受け、不正なソフトウェアのインストールを試みる事件が発生しました^{*24}。この事件では、レジストリに登録されたネームサーバ情報をなんらかの方法で書き換え、攻撃者が用意した偽のWebサイトに誘導していたことから、JPCERT/CCやJPRSより注意喚起が行われています^{*25}。

1.3 インシデントサーベイ

1.3.1 DDoS攻撃

現在、一般企業のサーバに対するDDoS攻撃が、日常的に発生するようになっており、その内容は、状況により多岐にわたります。しかし、攻撃の多くは、脆弱性などの高度な知識を利用したものではなく、多量の通信を発生させて通信回線を埋めたり、サーバの処理を過負荷にしたりすることでサービスの妨害を狙ったものになっています。

■ 直接観測による状況

図-2に、2014年7月から9月の期間にIJ DDoSプロテクションサービスで取り扱ったDDoS攻撃の状況を示します。

ここでは、IJ DDoSプロテクションサービスの基準で攻撃と判定した通信異常の件数を示しています。IJでは、ここに示す以外のDDoS攻撃にも対処していますが、攻撃の実態を正確に把握することが困難なため、この集計からは除外しています。

DDoS攻撃には多くの攻撃手法が存在し、攻撃対象となった環境の規模(回線容量やサーバの性能)によって、その影響度が異なります。図-2では、DDoS攻撃全体を、回線容

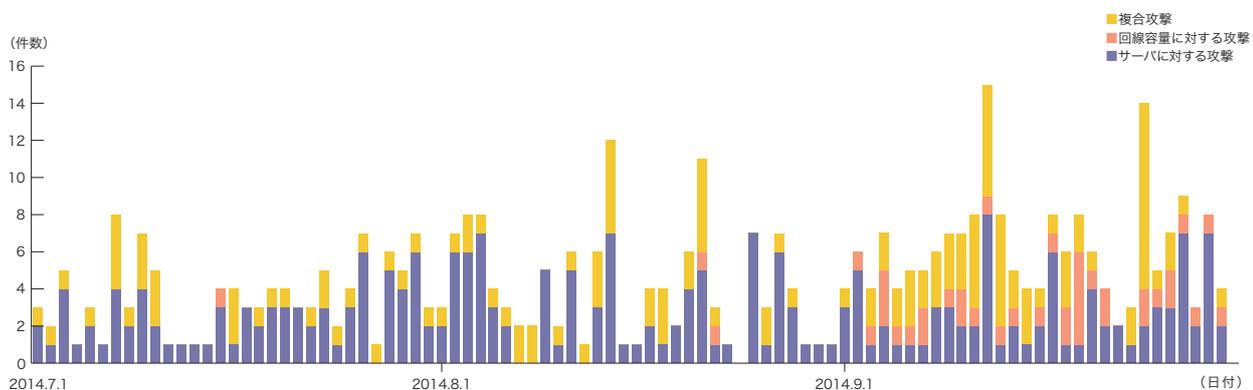


図-2 DDoS攻撃の発生件数

*23 総務省、「『電気通信事業におけるサイバー攻撃への適正な対処の在り方に関する研究会第一次とりまとめ』及び意見募集の結果の公表」(http://www.soumu.go.jp/menu_news/s-news/01ryutsu03_02000074.html)。

*24 Volexity Blog, "Democracy in Hong Kong Under Attack" (<http://www.volexity.com/blog/?p=33>)。

*25 株式会社日本レジストリサービス(JPRS)、「(緊急)登録情報の不正書き換えによるドメイン名ハイジャックとその対策について(2014年11月5日公開)」(<http://jprs.jp/tech/security/2014-11-05-unauthorized-update-of-registration-information.html>)。

量に対する攻撃^{*26}、サーバに対する攻撃^{*27}、複合攻撃(1つの攻撃対象に対し、同時に数種類の攻撃を行うもの)の3種類に分類しています。

この3ヵ月間でIJは、340件のDDoS攻撃に対処しました。1日あたりの対処件数は3.7件で、平均発生件数は前回のレポート期間と比べて減少しました。DDoS攻撃全体に占める割合は、サーバに対する攻撃が71.5%、複合攻撃が16.8%、回線容量に対する攻撃が11.8%でした。

今回の対象期間で観測された中で最も大規模な攻撃は、複合攻撃に分類したもので、最大48万6千ppsのパケットによって4.88Gbpsの通信量を発生させる攻撃でした。

攻撃の継続時間は、全体の90.9%が攻撃開始から30分未満で終了し、9.1%が30分以上24時間未満の範囲に分布しており、24時間以上継続した攻撃はありませんでした。なお、今回もっとも長く継続した攻撃は、複合攻撃に分類されるもので17時間36分にわたりました。

また、毎年この期間では、歴史的日付の前後でDDoS攻撃が多く見られます。9月に入ってからDDoS攻撃が増加しており、攻撃の傾向も変化が見られましたが、組織的な攻撃ではないことから、関連は確認できませんでした。

攻撃元の分布としては、多くの場合、国内、国外を問わず非常に多くのIPアドレスが観測されました。これは、IPスプーフィング^{*28}の利用や、DDoS攻撃を行うための手法としてのボットネット^{*29}の利用によるものと考えられます。

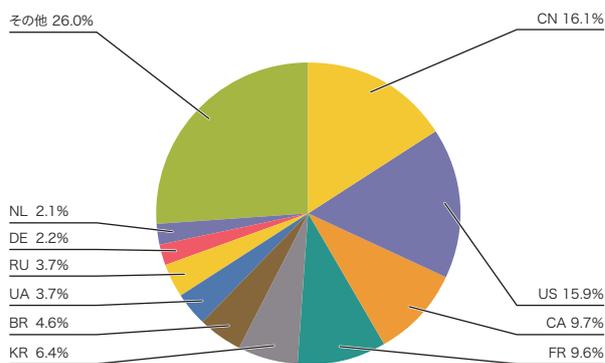


図-3 DDoS攻撃のbackscatter観測による攻撃先の国別分類

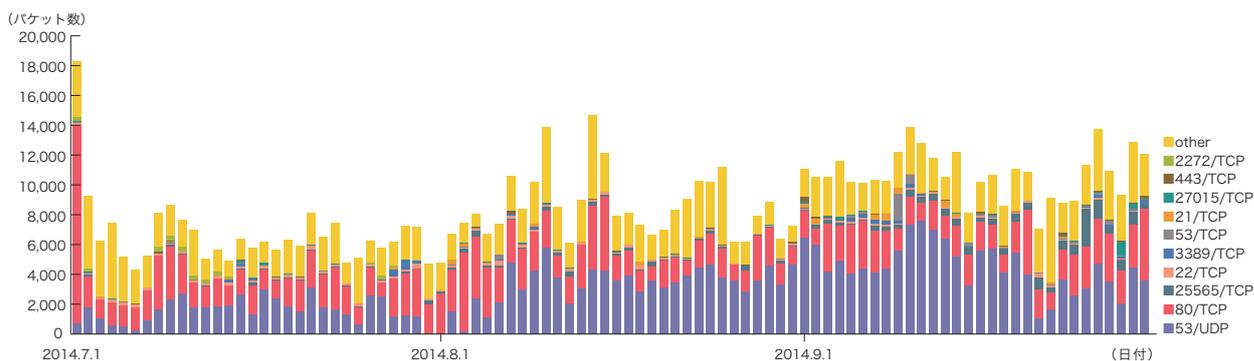


図-4 DDoS攻撃によるbackscatter観測(観測パケット数、ポート別推移)

*26 攻撃対象に対し、本来不必要な大きなサイズのIPパケットやその断片を大量に送りつけることで、攻撃対象の接続回線の容量を圧迫する攻撃。UDPパケットを利用した場合にはUDP floodと呼ばれ、ICMPパケットを利用した場合にはCMP floodと呼ばれる。

*27 TCP SYN floodやTCP connection flood、HTTP GET flood攻撃など。TCP SYN flood攻撃は、TCP接続の開始の呼を示すSYNパケットを大量に送付することで、攻撃対象に大量の接続の準備をさせ、対象の処理能力やメモリなどを無駄に利用させる。TCP Connection flood攻撃は、実際に大量のTCP接続を確立させる。HTTP GET flood攻撃は、Webサーバに対しTCP接続を確立した後、HTTPのプロトコルコマンドGETを大量に送付することで、同様に攻撃対象の処理能力やメモリを無駄に消費させる。

*28 発信元IPアドレスの詐称。他人からの攻撃に見せかけたり、多人数からの攻撃に見せかけたりするために、攻撃パケットの送出時に、攻撃者が実際に利用しているIPアドレス以外のアドレスを付与した攻撃パケットを作成、送付すること。

*29 ボットとは、感染後に外部のC&Cサーバからの命令を受けて攻撃を実行するマルウェアの一種。ボットが多数集まって構成されたネットワークをボットネットと呼ぶ。

■ backscatterによる観測

次に、IJJでのマルウェア活動観測プロジェクトMITFのハニーポット^{*30}によるDDoS攻撃のbackscatter観測結果を示します^{*31}。backscatterを観測することで、外部のネットワークで発生したDDoS攻撃の一部を、それに介在することなく第三者として検知できます。

2014年7月から9月の間に観測したbackscatterについて、発信元IPアドレスの国別分類を図-3に、ポート別のパケット数推移を図-4にそれぞれ示します。

観測されたDDoS攻撃の対象ポートのうち最も多かったものはDNSで利用される53/UDPで、対象期間における全パケット数の36.2%を占めています。次いでWebサービスで利用される80/TCPが27.3%を占めており、上位2つで全体の63.5%に達しています。またDNSに利用される53/TCP、SSHで利用される22/TCP、リモートデスクトップで利用される3389/TCP、FTPで利用される21/TCP、HTTPSで利用される443/TCPへの攻撃、通常は利用されない25565/TCPや27015/TCP、2272/TCPなどへの攻撃が観測されています。

今年2月から増加傾向にある53/UDPのbackscatterは、今回の期間も増加を続け、観測パケット数の最も多いポートになりました。これらのパケットのほとんどは「DNS水責め攻撃(Water Torture)^{*32}」と呼ばれる攻撃手法の特徴を持っています。また、観測されたパケットの発信元アドレスは広範にわたっています。

特に多くのbackscatterを観測した場合について、攻撃先のポート別にみると、Webサーバ(80/TCP)への攻撃としては、7月23日から8月15日にかけてウクライナのテレビ局、7月1日から24日にかけてロシアのホスティング事業者のサーバに対する攻撃をそれぞれ観測しています。特に後者は前回対象期間中の6月16日から攻撃が継続しています。また、9月に入って25565/TCPへの攻撃が多く観測されています。このポートは、特定のゲームのサーバで使われることがあります。攻撃対象は広範にわたっており、ロシアのホスティング事業者が持つ複数のサーバが攻撃される様子などが観測されています。9月9日から10日にかけて、パキスタンのドメインである.pkゾーンを担う複数のDNSサーバに対するDNS(53/TCP)への攻撃を観測しています。

また、今回の対象期間中に話題となったDDoS攻撃のうち、IJJのbackscatter観測で検知した攻撃としては、7月から8月初めにかけてAnonymousによる複数のイスラエル政府関連サイトへの攻撃を観測しています。

1.3.2 マルウェアの活動

ここでは、IJJが実施しているマルウェアの活動観測プロジェクトMITF^{*33}による観測結果を示します。MITFでは、一般利用者と同様にインターネットに接続したハニーポット^{*34}を利用して、インターネットから到着する通信を観測しています。そのほとんどがマルウェアによる無作為に宛先を選んだ通信か、攻撃先を見つけるための探索の試みであると考えられます。

*30 IJJのマルウェア活動観測プロジェクトMITFが設置しているハニーポット。「1.3.2 マルウェアの活動」も参照。

*31 この観測手法については、本レポートのVol.8(http://www.ijj.ad.jp/development/iir/pdf/iir_vol08.pdf)の「1.4.2 DDoS攻撃によるbackscatterの観測」で仕組みとその限界、IJJによる観測結果の一部について紹介している。

*32 Secure64 Software Corporation、「Water Torture: A Slow Drip DNS DDoS Attack」(<https://blog.secure64.com/?p=377>)。日本語での解説としては、株式会社日本レジストリサービス 森下氏による次の資料が詳しい。「DNS水責め(Water Torture)攻撃について」(http://2014.seccn.jp/dns/dns_water_torture.pdf)。

*33 Malware Investigation Task Forceの略。MITFは2007年5月から開始した活動で、ハニーポットを用いてネットワーク上でマルウェアの活動の観測を行い、マルウェアの流行状況を把握し、対策のための技術情報を集め、対策につなげる試み。

*34 脆弱性のエミュレーションなどの手法で、攻撃を受けつけて被害に遭ったふりをし、攻撃者の行為やマルウェアの活動目的を記録する装置。

■ 無作為通信の状況

2014年7月から9月の期間中に、ハニーポットに到着した通信の発信元IPアドレスの国別分類を図-5に、その総量(到着パケット数)の推移を図-6に、それぞれ示します。MITFでは、数多くのハニーポットを用いて観測を行っていますが、ここでは1台あたりの平均を取り、到着したパケットの種類(上位10種類)ごとに推移を示しています。また、この観測では、MSRPCへの攻撃のような特定のポートに複数回の接続を伴う攻撃は、複数のTCP接続を1回の攻撃と数えるように補正しています。

ハニーポットに到着した通信の多くは、Microsoft社のOSで利用されているTCPポートに対する探索行為でした。また、同社のSQL Serverで利用される1433/TCP、SSHで利用される22/TCP、DNSで利用される53/UDP、Telnetで利用される23/TCP、HTTP Proxyで用いられる8080/TCPに対する探査行為も観測されています。

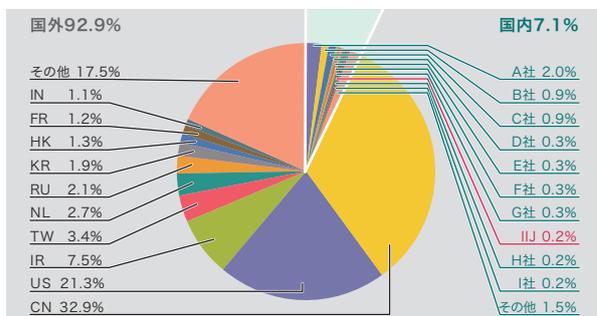


図-5 発信元の分布(国別分類、全期間)

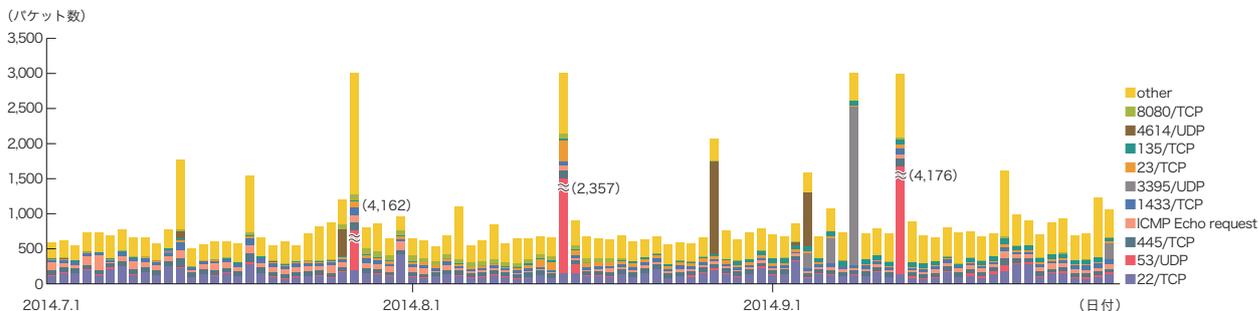


図-6 ハニーポットに到着した通信の推移(日別・宛先ポート別・1台あたり)

*35 MITFハニーポットはDNSの問い合わせパケットを受信しても、権威サーバやキャッシュサーバに問い合わせに行かないため、攻撃には加担していない。

*36 ここでは、ハニーポットなどで取得したマルウェアを指す。

*37 様々な入力に対して一定長の出力をす一方方向関数(ハッシュ関数)を用いて得られた値。ハッシュ関数は異なる入力に対しては可能な限り異なる出力を得られるよう設計されている。難読化やパディングなどにより、同じマルウェアでも異なるハッシュ値を持つ検体を簡単に作成できてしまうため、ハッシュ値で検体の一意性を保証することはできないが、MITFではこの事実を考慮した上で指標として採用している。

期間中、7月27日、8月14日、9月12日に53/UDPの通信が大量に発生しています。これらのパケットのほとんどはbackscatterによる観測でも触れた「DNS水責め攻撃(Water Torture)」のパケットを受信したためです*35。主に米国、カナダ、中国などに割り当てられた大量のIPアドレスから各ハニーポットに対して1回から数回程度の少ない回数問い合わせが行われています。この傾向から、攻撃者はポットネットなどを使ったと推測されます。問い合わせ内容は「ランダム文字列.実在するドメイン」のAレコードの解決要求でした。また9月8日にイランに割り当てられた1つのIPアドレスから、特定のハニーポットのIPアドレスに対して3395/UDPに対する通信が行われています。この通信の調査を行ったところ、長さは数十から数百バイトのランダムなデータが送信されていました。

■ ネットワーク上でのマルウェアの活動

同じ期間中でのマルウェアの検体取得元の分布を図-7に、マルウェアの総取得検体数の推移を図-8に、そのうちのユニーク検体数の推移を図-9にそれぞれ示します。このうち図-8と図-9では、1日あたりに取得した検体*36の総数を総取得検体数、検体の種類をハッシュ値*37で分類したものをユニーク検体数としています。また、検体をウイルス対策ソフトで判別し、上位10種類の内訳をマルウェア名称別に色分けして示しています。なお、図-8と図-9は前回同様に複数のウイルス対策ソフトウェアの検出名によりConficker判定を行いConfickerと認められたデータを除いて集計しています。

期間中の1日あたりの平均値は、総取得検体数が93、ユニーク検体数が20でした。未検出の検体をより詳しく調査した結果、シンガポールとフィリピンに割り当てられたIPアドレスからパスワードを盗み出すマルウェアなどが観測されました。また、未検出の検体の約54%がテキスト形式でした。これらテキスト形式の多くは、HTMLであり、Webサーバからの404や403によるエラー応答であるため、古

いワームなどのマルウェアが感染活動を続けているものの、新たに感染させたPCが、マルウェアをダウンロードしに行くダウンロード先のサイトが既に閉鎖させられていると考えられます。

MITF独自の解析では、今回の調査期間中に取得した検体は、ワーム型96.0%、ダウンロード型4.0%でした。また解析により、1個のボットネットC&Cサーバ^{*38}と16個のマルウェア配布サイトの存在を確認しました。

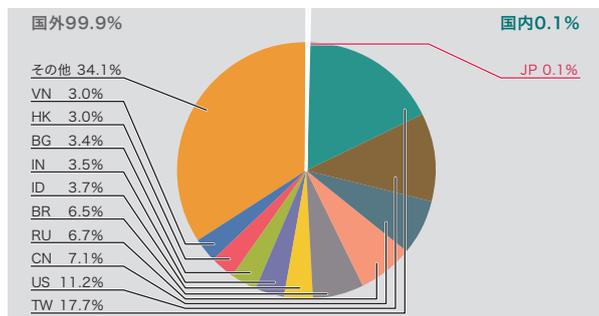


図-7 検体取得元の分布(国別分類、全期間、Confickerを除く)

■ Confickerの活動

本レポート期間中、Confickerを含む1日あたりの平均値は、総取得検体数が25,435、ユニーク検体数は672でした。短期間での増減を繰り返しながらも、総取得検体数で99.6%、ユニーク検体数で97.0%を占めています。このように、今回の対象期間でも支配的な状況が変わらないことから、Confickerを含む図は省略しています。本レポート期間中の総取得検体数は前回の対象期間と比較し、約20%減少

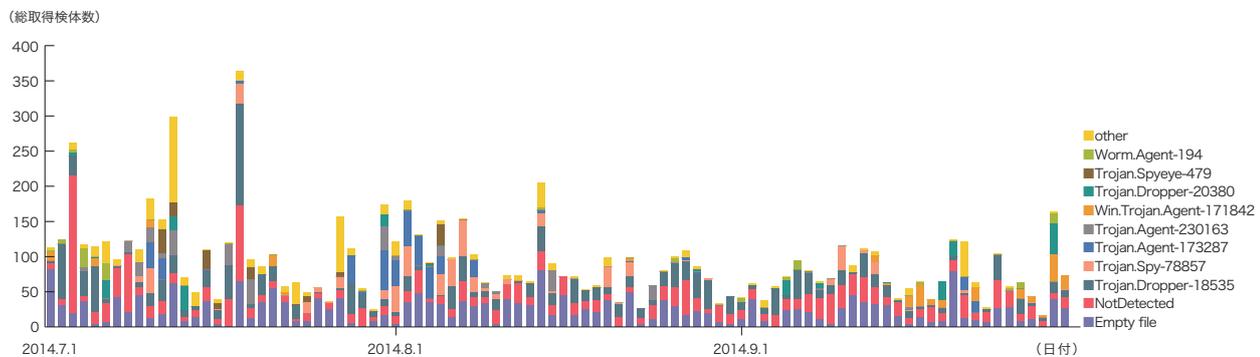


図-8 総取得検体数の推移(Confickerを除く)

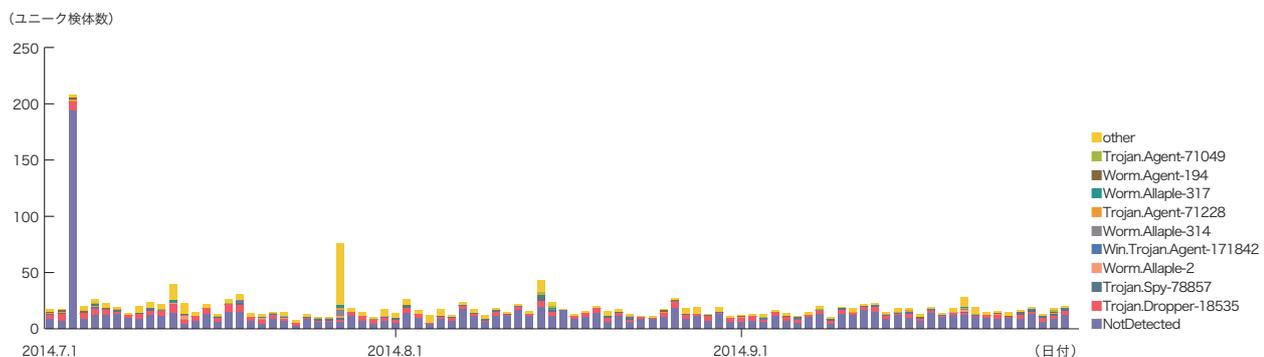


図-9 ユニーク検体数の推移(Confickerを除く)

*38 Command & Controlサーバの略。多数のボットで構成されたボットネットに指令を与えるサーバ。

しています。また、ユニーク検体数は前号から約7%減少しました。Conficker Working Groupの観測記録^{*39}によると、2014年9月30日現在で、ユニークIPアドレスの総数は1,026,417とされています。2011年11月の約320万台と比較すると、約32%に減少したことになりますが、依然として大規模に感染し続けていることが分かります。

1.3.3 SQLインジェクション攻撃

IJでは、Webサーバに対する攻撃のうち、SQLインジェクション攻撃^{*40}について継続して調査を行っています。SQLインジェクション攻撃は、過去にも度々流行し話題となった攻撃です。SQLインジェクション攻撃には、データを盗むための試み、データベースサーバに過負荷を起すための試み、コンテンツ書き換えの試みの3つがあることが分かっています。

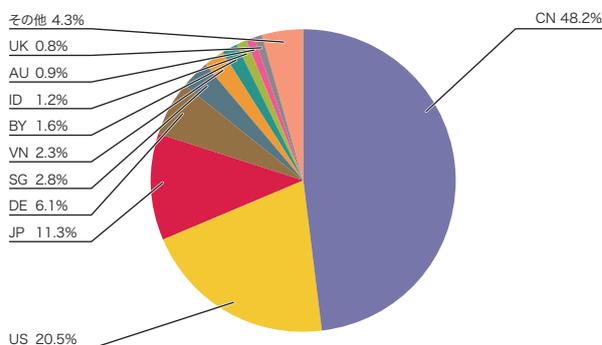


図-10 SQLインジェクション攻撃の発信元の分布

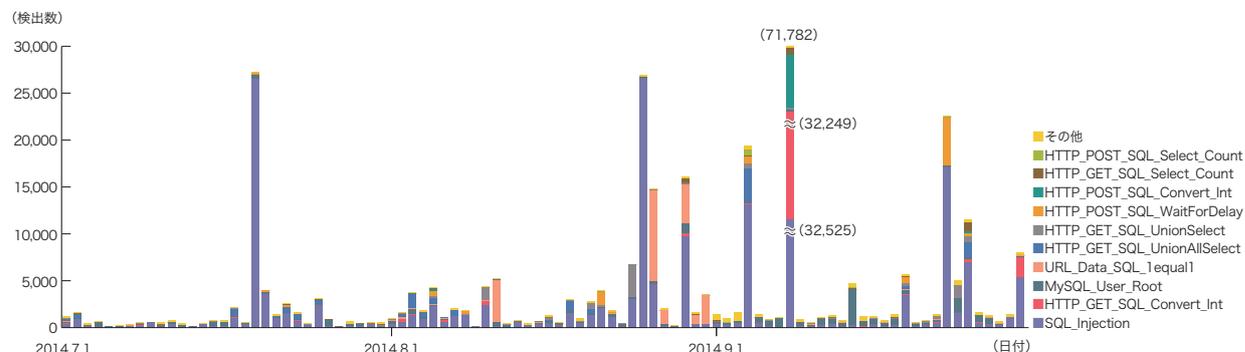


図-11 SQLインジェクション攻撃の推移(日別、攻撃種類別)

2014年7月から9月までに検知した、Webサーバに対するSQLインジェクション攻撃の発信元の分布を図-10に、攻撃の推移を図-11にそれぞれ示します。これらは、IJマネージドIPSサービスのシグネチャによる攻撃の検出結果をまとめたものです。

発信元の分布では、中国48.2%、米国20.5%、日本11.3%となり、以下その他の国々が続いています。Webサーバに対するSQLインジェクション攻撃の発生件数は前回に比べて大幅に増加しました。これは中国を発信元とする攻撃が大幅に増えたためです。

この期間中、7月19日には、中国の特定の攻撃元より特定の攻撃先に対する攻撃が発生しています。8月25日には米国の複数の攻撃元より、特定の攻撃先に対する攻撃が発生しています。9月8日には中国の特定の攻撃元より、特定の攻撃先への大規模な攻撃が発生していました。9月23日には、中国の特定の攻撃元から特定の攻撃先に対する攻撃が発生していました。これらの攻撃はWebサーバの脆弱性を探る試みであったと考えられます。

ここまで示したとおり、各種の攻撃はそれぞれ適切に検出され、サービス上の対応が行われています。しかし、攻撃の試みは継続しているため、引き続き注意が必要な状況です。

*39 Conficker Working Groupの観測記録(<http://www.confickerworkinggroup.org/wiki/pmwiki.php/ANY/InfectionTracking>)。

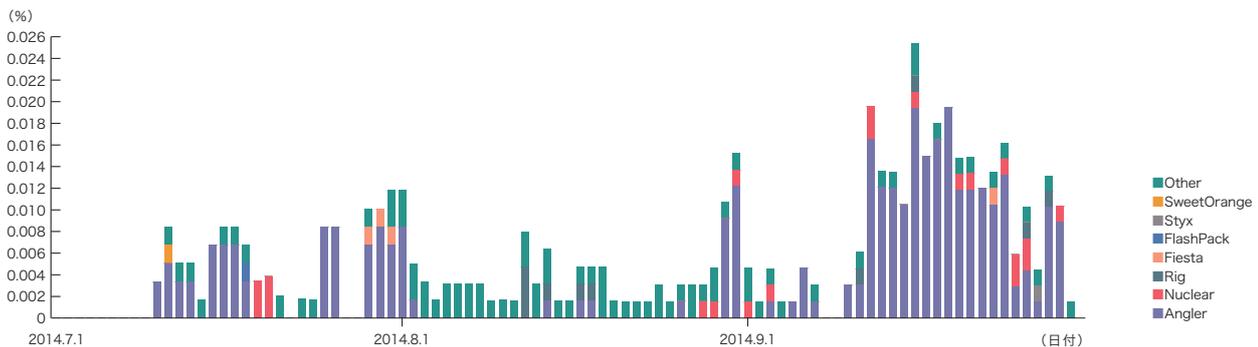
*40 Webサーバに対するアクセスを通じて、SQLコマンドを発行し、その背後にいるデータベースを操作する攻撃。データベースの内容を権限なく閲覧、改ざんすることにより、機密情報の入手やWebコンテンツの書き換えを行う。

1.3.4 Webサイト改ざん

MITFのWebクローラ(クライアントハニーポット)によって調査したWebサイト改ざん状況を示します*41。このWebクローラは国内の著名サイトや人気サイトなどを中心とした数万のWebサイトを日次で巡回しており、更に巡回対象を順次追加しています。また、一時的にアクセス数が増加したWebサイトなどを対象に、一時的な観測も行っています。一般的な国内ユーザによる閲覧頻度が高いと考えられるWebサイトを巡回調査することで、改ざんサイトの増減や悪用される脆弱性、配布されるマルウェアなどの傾向が推測しやすくなります。

2014年7月から9月の期間は、4月から6月の期間に比べると、約2倍程度の頻度でドライブバイダウンロードが観測されています(図-12)。攻撃の内訳は、前期間と同じくAngler及びNuclearによる攻撃が多くを占めており、これらのExploit Kitは、いずれもJavaやFlash、Silverlightなどのプラグインの脆弱性を悪用する機能を備えています。特にAnglerは精力的に機能追加が行われており、8月下旬には感染時にクライアントのディスク上にマルウェアファイルを保存させないことでアンチウイルスソフトウェアによる検知の回避を試みる仕組みが確認されました*42。

また、9月中旬にWebクローラシステムの攻撃検出口ジックを一部改良したところ、Anglerの観測数が急増しました。



*調査対象は日本国内の数万サイト。近年のドライブバイダウンロードは、クライアントのシステム環境やセッション情報、送信元アドレスの属性、攻撃回数などのノルマ達成状況などによって攻撃内容や攻撃の有無が変わるよう設定されているため、試行環境や状況によって大きく異なる結果が得られる場合がある。
*7月1日～7月7日はWebクローラを停止していたため、攻撃を検知していない。

図-12 Webサイト閲覧時のドライブバイダウンロードの発生率(%) (Exploit Kit別)

*41 Webクローラによる観測手法については、本レポートのVol.22 (http://www.ij.ad.jp/company/development/report/iir/pdf/iir_vol22.pdf)の「1.4.3 WebクローラによるWebサイト改ざん調査」で仕組みを紹介している。

*42 同機能については、Malware don't need Coffee「Angler EK: now capable of "fileless" infection (memory malware)」(<http://malware.dontneedcoffee.com/2014/08/angler-ek-now-capable-of-fileless.html>)で詳解されている。

*43 Rigについては、Kahu Security「RIG Exploit Pack」(<http://www.kahusecurity.com/2014/rig-exploit-pack/>)で詳解されている。

このことから、一時的にAnglerが消滅しているように見える8月中も、実は同Exploit Kitによる攻撃が継続していた可能性が高いと推測されます。

新たな傾向として、8月中旬ごろから、Rigが観測され始めました。これは比較的新しいExploit Kitで、前述のAngler、Nuclearと同じくJavaやFlash、Silverlightの脆弱性を攻撃する機能を備えており、有償のサービスとして提供されています*43。なお、Rigによる攻撃が観測された一部のケースでは、1カ所の誘導元Webサイトから複数のExploit Kitへの誘導が観測されました。

一方、改ざんされ、誘導元となっているWebサイトに関しては、比較的低知名度の低いWebサイトなどで、2～4週間程度改ざんされた状態が継続するケースが多数確認されており、改ざん原因の調査や根本的な対策が行われていないケースが後を絶たない状況であることが窺えます。

全体として、ドライブバイダウンロードの発生率が増加傾向となっているものと推測される状況です。Webサイト運営者はWebコンテンツの改ざん対策、閲覧者側はブラウザや関連プラグインなどの脆弱性対策を徹底し、注意を継続することを推奨します。

1.4 フォーカスリサーチ

インターネット上で発生するインシデントは、その種類や規模が時々刻々と変化しています。このため、IJでは、流行したインシデントについて独自の調査や解析を続けることで対策につなげています。ここでは、これまでに実施した調査のうち、Bashの脆弱性Shellshockについて、POODLE attack、リスト型攻撃の発生状況と対策の3つのテーマについて紹介します。

1.4.1 Bashの脆弱性Shellshockについて

■ Shellshockとは

この脆弱性CVE-2014-6271^{*44}は、2014年9月24日にBash^{*45}の対策バージョンと共に公開されました。脆弱性の影響としてはリモートから任意のコード実行が可能です。Bashは通常ローカルで使用されるシェルプログラムであるため、リモートからの攻撃とは無縁と思われがちですが、その影響が明らかになるにつれて大きな騒ぎになりました。

脆弱性公開と同時に対策バージョンがリリースされましたが、公開後から約1週間で、関連する複数の新しい脆弱性が立て続けに公開されました。表-1が関連する脆弱性の一覧です。CVE-2014-7169^{*46}は元々の修正が不十分であっ

たとして新しい脆弱性として登録されました。CVE-2014-7186^{*47}、CVE-2014-7187^{*48}は修正の過程で発見された脆弱性です。CVE-2014-6277^{*49}、CVE-2014-6278^{*50}はパッチがアップストリームに取り込まれる過程で作り込まれました。Shellshockによる攻撃を防ぐためには、これらの脆弱性すべてに対応する必要があります。

■ 脆弱性への対応

Bashはシェルプログラムという性質上、サーバプログラムやスクリプト言語のインタプリタなどとは異なり、最新版の機能が必要とされる機会はあまりありません。そのため、多くの場合で各ディストリビューションの提供するバイナリパッケージを使用しています。Shellshockには複数の脆弱性が関連していますが、アップストリームに取り込まれた修正のみ影響を受ける脆弱性もあるため、適用されているパッチの状態により対応が必要な脆弱性は異なります。

ディストリビューションの例として、CentOS^{*51}の各種リリースにおける各脆弱性の対応状況を表-2に示します。CVE-2014-6271の対策版において、修正が不十分であった点は他と変わりませんが、それ以降の脆弱性においては同時に対応し、アップストリームの修正に起因するCVE-2014-6277、CVE-2014-6278については影響を受けていません。

表-1 Shellshock関連脆弱性一覧

CVE ID	公開日時	影響	備考
CVE-2014-6271	2014年9月24日	任意コード実行	発端となる脆弱性
CVE-2014-7169	2014年9月25日	任意コード実行	CVE-2014-6271の修正不十分に起因
CVE-2014-7186	2014年9月26日	DoS	-
CVE-2014-7187	2014年9月26日	DoS	-
CVE-2014-6277	2014年9月27日	DoS	アップストリーム版の修正に起因
CVE-2014-6278	2014年9月27日	任意コード実行	アップストリーム版の修正に起因

表-2 CentOS各リリースにおける対策版bashパッケージ

CVE ID	CentOS5系	CentOS6系	CentOS7系
CVE-2014-6271	bash-3.2-33.el5.1	bash-4.1.2-15.el6_5.1	bash-4.2.45-5.el7_0.2
CVE-2014-7169			
CVE-2014-7186	bash-3.2-33.el5_10.4	bash-4.1.2-15.el6_5.2	bash-4.2.45-5.el7_0.4
CVE-2014-7187			
CVE-2014-6277	対策不要	対策不要	対策不要
CVE-2014-6278	対策不要	対策不要	対策不要

*44 CVE-2014-6271 bash: specially-crafted environment variables can be used to inject shell commands(https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2014-6271).

*45 GNU Bash: The GNU Bourne-Again SHell(<http://www.gnu.org/software/bash/>).

*46 CVE-2014-7169 bash: code execution via specially-crafted environment Incomplete fix for CVE-2014-6271 (https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2014-7169).

*47 CVE-2014-7186 bash: parser can allow out-of-bounds memory access while handling `redir_stack` (https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2014-7186).

*48 CVE-2014-7187 bash: off-by-one error in deeply nested flow control constructs(https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2014-7187).

*49 CVE-2014-6277 bash: uninitialized here document closing delimiter pointer use(https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2014-6277).

*50 CVE-2014-6278 bash: incorrect parsing of function definitions with nested command substitutions(https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2014-6278).

*51 CentOS Project(<http://www.centos.org/>).

これはCentOSの元となるRed Hat Enterprise Linux^{*52}がBashの最新版へのアップデートではなく、ベースとなるパッケージに対して、脆弱性を修正するパッチを当てて対応しているためです。他のディストリビューションにおいても同様に、修正方法により影響有無が分かれています。

■ 脆弱性の影響

ローカルシェルとして使用されるBashにおける脆弱性が、何故リモートから悪用可能な脆弱性へ発展したのでしょうか。今回の脆弱性は、特定の環境変数の値が設定された状態において、Bashプログラムが起動されると影響を受けるといえるのです。Bashがシェル関連の呼び出しで使われる/bin/shを兼ねている環境もあり、こちらも同様の影響を受けます。

実際にこの脆弱性を用いた攻撃は既に観測されており、最も狙われている箇所はCGI^{*53}です。分かりやすい影響例として、Webサーバ上で動作するBashで書かれたCGIプログラムがありますが、実際の環境においてはほぼ利用されていないと考えられます。しかしながら、他の言語で作られたCGIプログラムであったとしても、内部的にシェルに依存しているために影響を受ける場合があります。

また、アプライアンス製品や組み込み機器の制御OSとしてもLinuxが多く採用されています。身近な物としてはロードバランサ、ホームルータ、ファイルサーバなどが挙げられます。これらにもBashが組み込まれていたり、管理画面においてCGIが使われていたり、普通のUNIXサーバ同様の影響を受けてしまう場合もあります。これらを踏まえて、今回

の脆弱性の影響を理解するにあたり、UNIX環境におけるプログラムの実行と、CGIの処理の流れについて説明します。

■ UNIX環境におけるプログラム実行

UNIX環境においては、Windows環境とは異なり完全新規のプロセスを作成することができず、親プロセスの複製(fork)を経て、子プロセスを実行対象のプロセスへ変化(exec)させる手順を辿ります。最終的には子プロセスは実行対象のプログラムへと変化して親プロセスとは別物になりますが、複製を経由するため一部のデータが引き継がれます。図-13にこの処理の流れを示します。

変化する際に呼び出すexec関数には表-3に示すとおり複数の種類があり、今回の脆弱性において鍵となる環境変数の扱いが異なります。呼び出し時に明示的に環境変数を指定しないexec関数においては、親プロセスから子プロセスに引き継がれた環境変数が、そのまま実行対象のプロセスへ変化した後も使用されます。

■ CGIの処理の流れ

CGIはWebサーバ経由で動的コンテンツを提供する仕組みです。Webアプリケーションフレームワークの普及に伴い、以前よりは少なくなっていますが依然として使われています。WebサーバはHTTP経由で受け取ったヘッダを、環境変数へ展開してCGIプログラムを起動します。この仕様が今回の脆弱性において、CGIが最も影響を受けやすい箇所となっている理由です。CGIプログラムとして直接Bashが起動されない限り、この時点では影響を受けません。ですが、

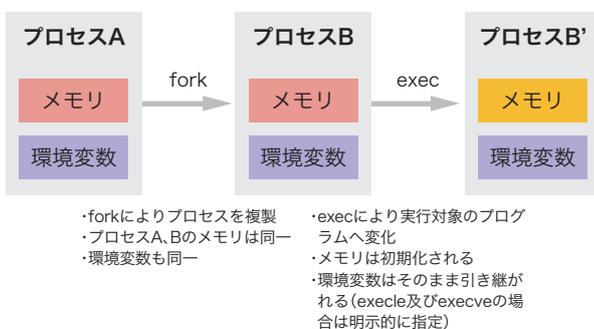


図-13 UNIX環境における外部プログラム実行

表-3 exec系関数の動作

関数名	環境変数の扱い
execl	引き継ぐ
execle	明示的に指定
execlp	引き継ぐ
execv	引き継ぐ
execve	明示的に指定
execvp	引き継ぐ

*52 Red Hat Enterprise Linux (<http://www.redhat.com/en/technologies/linux-platforms/enterprise-linux>).

*53 RFC3875 The Common Gateway Interface Version 1.1 (<http://www.ietf.org/rfc/rfc3875.txt>).

前述のとおり環境変数は子プロセスに引き継がれていくため、他の言語で書かれていたとしてもシェルを経由して外部プログラムを呼び出してしまうと、影響を受ける可能性があります。直接呼び出したプログラムに限らず、ライブラリやモジュールを経由して、間接的に呼び出されたプログラムであったとしても同様です。図-14にCGI経由のプログラム呼び出しにおける影響の有無を示します。

また、スクリプト言語においてはプログラムの記述方法により、意図せずにシェルが呼び出されてしまうこともあります。スクリプト言語の例としてPerl^{*54}における動作を示したものが表-4です。このように、同一の関数による呼び出しであっても、引数次第で暗黙的にシェル経由の呼び出しへ展開されてしまいます。環境変数はすべて引き継がれていますので、シェル経由の呼び出しへ展開される場合は、今回の脆弱性の影響を受けます。

■ まとめ

これまで説明したように、Bashという一見リモートからの攻撃とは無縁に見えるプログラムの脆弱性であったとして

も、利用のされ方により、外部からこの脆弱性狙った攻撃にさらされることになります。

このような影響条件は、ソースコードを元にした静的な調査などではすべての実行パスを網羅的に調査することは困難です。一方、Bashのアップデートはサーバプログラムなどとは異なり、多くの場合において再起動を必要としないため、少しでも影響が疑われる箇所であれば速やかに対策版へ更新することをお勧めします。

アプライアンス製品や組み込み機器については、ベンダーによるファームウェア修正が必要になります。対策版ファームウェアのリリースに時間がかかる製品もありますので、その場合は影響が疑われる箇所へのアクセスを制限するなどの暫定的な対策も有効です。

脆弱性への対策としては、回避策の適用、対策版への更新、セキュリティ装置による防御などの様々な手段があります。対象のシステム構成により、最適な対策は異なりますので、その都度判断して適切に対処することが重要です。

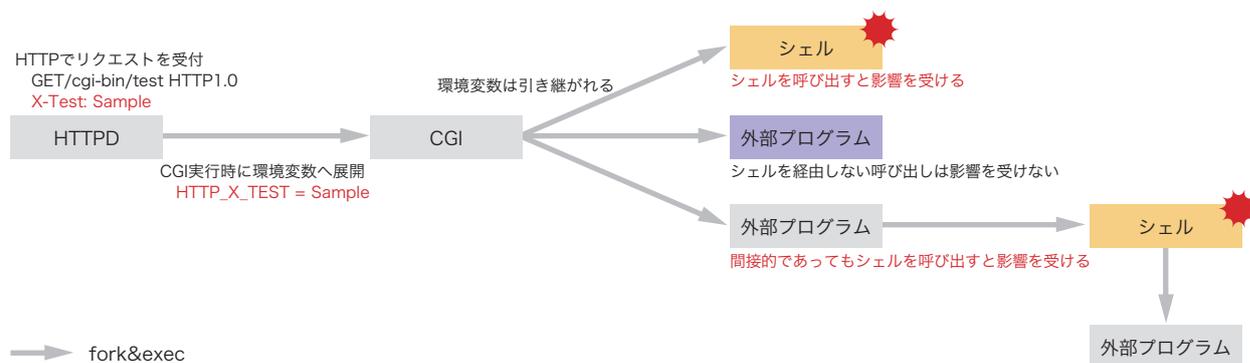


図-14 CGIにおける影響

表-4 Perlにおける外部プログラム呼び出し

Perlコード	exec関数への展開	シェル呼び出し	環境変数
<code>`/bin/lis`</code>	<code>execvp("/bin/lis", ["/bin/lis"])</code>	なし	引き継ぐ
<code>`/bin/lis 2>/dev/null`</code>	<code>execl("/bin/sh", ["-c", "/bin/lis 2>/dev/null"])</code>	あり	引き継ぐ
<code>system("/bin/lis")</code>	<code>execvp("/bin/lis", ["/bin/lis"])</code>	なし	引き継ぐ
<code>system("/bin/lis &")</code>	<code>execl("/bin/sh", ["-c", "/bin/lis &"])</code>	あり	引き継ぐ
<code>open(FH, "/bin/lis ")</code>	<code>execvp("/bin/lis", ["/bin/lis"])</code>	なし	引き継ぐ
<code>open(FH, "/bin/lis 2>/dev/null ")</code>	<code>execl("/bin/sh", ["-c", "/bin/lis 2>/dev/null"])</code>	あり	引き継ぐ

*54 The Perl Programming Language(<https://www.perl.org/>)。

1.4.2 POODLE attack

米国時間2014年10月14日に、Googleの研究チームからSSLv3^{*55}への新たな攻撃が公開されました^{*56}。POODLE attack (Padding Oracle On Downgraded Legacy Encryption attack) と呼ばれる本手法はBEAST攻撃^{*57}に類似した中間者攻撃で、ブラウザから大量のリクエストをサーバに送りつけることによるトライ&エラーを繰り返すことでSSLで暗号化された攻撃対象データを1バイトずつ復号する攻撃手法です。現実的な被害としては、Cookieの搾取が挙げられています。なお本手法は一見TLSv1.0^{*58}にも適用可能であるかのように見えますが、SSLv3.0とTLSv1.0のパディング手法の違いにより、TLSv1.0では現実的な攻撃にはなりません。一方でSSLv3.0では1バイトを盗み見るために最大255回のクエリをサーバに送りつけるだけで攻撃が可能となります。本節では、POODLE attackの技術的背景と本攻撃の現実性について取り上げます^{*59}。

■ CBCモードの概要とCBCモードが持つ潜在的な脆弱性

POODLE attackはBEAST攻撃と同じく暗号モードとしてCBC (Cipher Block Chaining) モードが利用されたときのみ成功します。RC4をはじめとしたストリーム暗号は、鍵情報をもとに発生させたキーストリームを平文にXOR演算で足しこむことで暗号化を行う方式であり、任意長のデータを暗号化することができます。一方で、DESやAESなどのブロック暗号は定められた入力長(ブロック長:DESは64ビット、AESは128ビット)のデータを暗号化アルゴリズムに入力し、入力長と同じブロック長の暗号文を得る方式です。通常ブロック暗号を利用する際には、ブロック長よりもはるかに長いデータを暗号化しますので、何度も逐次的に暗号化アルゴリズムで暗号処理を行う必要があります。1つ前のブロック暗号処理で得られたデータを次のデータ処理でどのように利用するかを定めた方式のことを暗号モードと呼び、いくつもの方式が提案されています。図-15はその1つであるCBC暗号モード^{*60}

の概要を示しています。まず初期値としてブロック長のIV (Initial Vector)^{*61}を用意します。平文をブロック長ごとに分割したときの最初のパートである P_1 にIVをXOR演算で足し合わせたデータに、実際のブロック暗号における暗号化処理を行い、その出力として暗号文 C_1 が得られます。次の平文パート P_2 を暗号化するには、前のブロック暗号化処理で得られた C_1 をXOR演算で足し合わせたデータを平文として入力して暗号化し、同様に C_2 を得ます。このように $C_i = \text{ENC}(P_i \oplus C_{i-1})$ という暗号化処理を繰り返すことで、平文全体の暗号化を計算します。逆に復号する際には $P_i = \text{DEC}(C_i) \oplus C_{i-1}$ という復号処理を逐次的に繰り返すことで、平文を復元することができます。

この暗号化処理において、ブロック長でちょうど割り切れないデータを扱う場合にはパディング処理が必要となります。これは、平文の最後のパートがブロック長に満たない場合、何がしかのデータでブロック長になるように埋めてブロック暗号化処理が行えるようにする処理です。TLSではPKCS#5

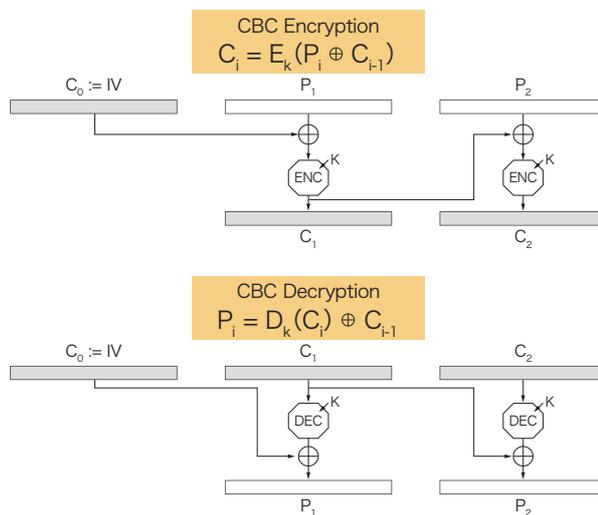


図-15 CBC暗号モードの概要

*55 IETF, "The SSL Protocol Version 3.0" (<http://tools.ietf.org/html/rfc6101>).

*56 Bodo Möller, Thai Duong, Krzysztof Kotowicz, "This POODLE Bites: Exploiting The SSL 3.0 Fallback" (<https://www.openssl.org/~bodo/ssl-poodle.pdf>).

*57 Thai Duong, Juliano Rizzo, "BEAST - Here Come The XOR Ninjas", 2011.

*58 IETF, "The TLS Protocol Version 1.0" (<http://tools.ietf.org/html/rfc2246>).

*59 本節は技術的な解説に重きを置いている。POODLE attackの概要については、以下に示す暗号プロトコル評価技術コンソーシアムによる速報を参照のこと。SSLv3仕様そのものに対するPOODLE attackについて (https://www.cellos-consortium.org/jp/index.php?PoodleAttack_20141015_J)。

*60 NIST, "NIST Special Publication 800-38A Recommendation for Block Cipher Modes of Operation: Methods and Techniques" (<http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf>).

*61 IV (Initial Vector) は秘密にする必要はない。ただし復号時には鍵と同様にIVが必要となる。

padding方式^{*62}が採用されており、図-16はブロック長が8バイト(例えばDES)の場合のパディングの例を表しています。(a)はブロック長に満たすまでに3バイト足りないためパディングデータとして3バイト分を0x03で埋めます。同様に(b)は5バイト足りないため5バイト分を0x05で埋めています。(c)はちょうどブロック長の倍数のデータを暗号化しようとするケースですが、平文のどこまでパディングされていたかの境界を復号時に認識できるようにするために「8バイト分(ちょうどブロック長)足りない」ということにして当該ブロックのすべてを0x08で埋めていきます。

一方でSSLv3.0では異なるパディング方式が採用されており、この違いが今回のPOODLE attackを生み出すきっかけの1つとなってしまいました。図-17はPKCS#5 paddingとSSLv3 paddingの違いを示しています。SSLv3で採用されているパディング方式では、パディングする最終バイトをPKCS#5 padding方式と同様に「パディングするバイト長」とし、それ以外のパディングデータはランダムデータを埋めていくと定められています。不運なことに、SSLv3でもTLSでもこのパディングされるエリアは、通信路でデータが改ざんされていないことを示すデータであるMAC(Message Authentication Code)の対象範囲には入っていません。そのため、パディングデータを改ざんされたとしてもMAC

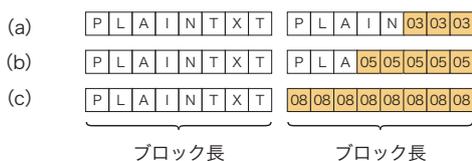
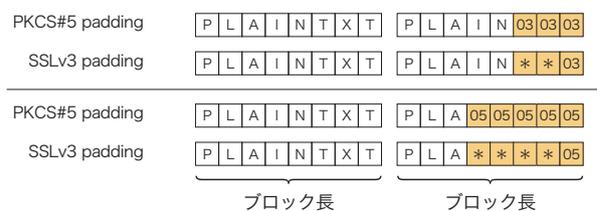


図-16 ブロック長が8バイトの場合のPKCS#5 paddingの例

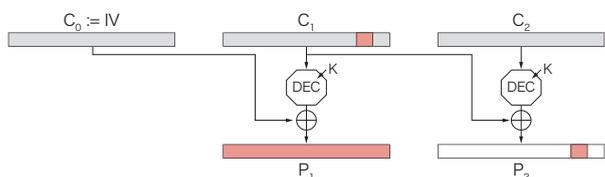


※色付けされたパディングデータにおける*はランダムなデータであることを示している。

図-17 PKCS#5 paddingとSSLv3 paddingの違い

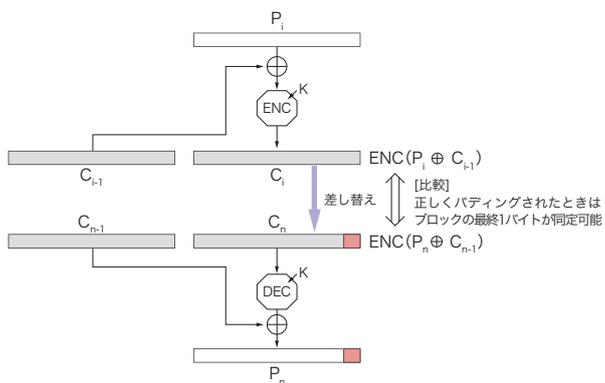
の検証では、改ざんを検知することができません。一方で、TLSにおいては上記説明したパディングデータの制約条件を用いて、パディング部分の改ざんを検知できる可能性はありますが、SSLv3では最終バイト以外のパディングデータを改ざんされたとしてもそれを検知することができません。POODLE attackはこのちょっとした隙間を狙った攻撃とも言えます。

図-18はCBCモードを利用した復号時において、暗号文が改ざんされた場合の波及範囲を示しています。この図において暗号文 C_i の一部を改ざんすることで平文 P_i はブロック全体が変更されてしまいますが、平文 P_2 は攻撃者の意図する箇所を自由に交換することができます。図-19はこの特徴を用いたPadding Oracle Attack^{*63}の原理を示しています。平文の長さが分かっているという前提で P_n までのデータが暗号化されている場合、最後のブロック C_n を中間者が「攻撃対象」、つまり復号したいデータである C_i に差し替えます。この



※暗号文の1バイトを改ざんして復号処理した場合には、当該ブロックの平文はブロック全体に渡って影響が及ぶが、次ブロックにおける平文において意図した箇所を改ざんすることが可能である。

図-18 CBCモードにおける暗号文が改ざんされた場合の波及範囲



※サーバのPaddingチェック機能においてPaddingが正しい場合にはアクセプトされるが誤っている場合にはエラーが返却される。

図-19 Padding Oracle Attackの原理

*62 IETF, "PKCS #5: Password-Based Cryptography Specification Version 2.0" (<http://tools.ietf.org/html/rfc2898>).

*63 SSL/TLSだけでなくSSHやIPsecにおいても同様の攻撃が公開されてきた。いずれも仕様で定められた「データ形式の制約」に則っているかどうかを、サーバからのエラー情報を基に判定して暗号化データを復号する攻撃手法である。M.R. Albrecht, K.G. Paterson, G.J. Watson, Plaintext Recovery Attacks Against SSH, 30th IEEE Symposium on Security and Privacy, 2009. (<http://www.isg.rhul.ac.uk/~kp/SandPfinal.pdf>). J.P. Degabriele and K.G. Paterson, On the (In)security of IPsec in MAC-then-Encrypt Configurations, Proceedings of the 17th ACM Conference on Computer and Communications Security, 2010. (<http://portal.acm.org/citation.cfm?id=1866363>).

とき平文の長さが既知のため、パディングされたデータ長が格納されている P_n の最終バイトは既知となります。SSL/TLSサーバは手順どおりパディングチェックを行い、もし仕様で定められたパディング方式に則っていない場合にはエラーを返却します。このエラー返却機能(Padding Oracle)を使うと、サーバが暗号化データをアクセプトした場合、パディング部分には正しいデータが格納されていることを意味します。SSLv3においては P_i の最終バイトと P_n の最終バイトが一致するときにアクセプトされますので、 $P_i \oplus C_{i-1}$ の最終バイトと $P_n \oplus C_{n-1}$ の最終バイトが一致することから P_i の最終バイトを(暗号化鍵を知らなくても)復元することができます。実際には、MACによるチェックが入るためパディングデータよりも前の平文部分が改ざんされたことが検知できますので、うまくいかないことがほとんどです。しかし、今回POODLE attackで指摘されたように、うまくそのブロックがパディングデータのみで埋め尽くされる状態(図-16の(c)のような状態)にすることでMACの影響範囲を最終ブロックから外すことができます。このとき最終の1バイトが合致しているか試行すると256回に1度だけ必ずアクセプトされます。一方でTLSにおいては、図-16(c)のように、ブロックすべてのバイトが同じデータで埋め尽くされる必要があります。この例ではブロック長が8バイトですから、 2^{64} 回の試行で1度だけ成功します。またAESではブロック長が16バイトですから 2^{128} 回の試行が必要なことから、TLSにおいてこの攻撃は現実的ではない潜在的問題点としてみなされてきました。しかし、今回指摘されたPOODLE Attackでは、1バイトを復元するために 2^8 程度の計算で済むため、現実的な攻撃として認識されました。

■ POODLE attackの現実性

SSHにおいては、CBCモード以外にも利用可能な暗号モードの1つであるCTRモードが利用可能であったため、CBC

モードの利用を取りやめることでPadding Oracle Attackを防ぐことができました。しかし、SSL/TLSにおいてはCTRモードの利用は仕様上定められていません。またSSLv3においては、仕様で規定されているCipherSuites(暗号アルゴリズム)において、輸出規制のためにかつて利用されていた40ビット鍵の強度しか持たないアルゴリズムやNULL(暗号化なし)以外のものでもCBCモードを利用しないときにはRC4を使わざるを得ません。しかしRC4には昨年新たな攻撃手法が立て続けに公開されており^{*64*65}、もはやRC4も脆弱なアルゴリズムであると認識されています。

POODLE attackと同様にPadding Oracle Attackに分類されているBEAST攻撃は、SSLv3及びTLSv1.0にて成功可能であり、TLSv1.1にて仕様としてこの脆弱性を回避する設計がなされています。具体的には前セッションの暗号化データを次セッションのIVとして利用する際の潜在的問題を回避し、新たにIVを作り直す工程が入りました。またBEAST攻撃が存在しながらもSSLv3及びTLSv1.0が現在も利用されてきたのは、1/n-1分割法と呼ばれる対策方法が有効であることが知られているためです^{*66}。この手法は主要ブラウザにおいて既に対策されていますが、POODLE attackに適用することはできません。今後、1/n-1分割法と同様にSSLv3の延命技術が考案される可能性はありますが、現時点ではSSLv3を安全に利用する方法が皆無となりました。

今回の攻撃に対する根本的な対策として、1)SSLv3を無効にする(クライアント、サーバのいずれか)、2)TLS_FALLBACK_SCSVの導入(クライアント、サーバ共に対策が必要)、の2つが挙げられています。1)に関しては、主要ブラウザは今後SSLv3を無効にするアナウンスされました^{*67*68*69}。また、ブラウザに自ら設定してSSLv3を無効化する方法も案内されています^{*70*71}。サーバサイドにおいてもSSLv3を無

*64 Takanori ISOBE, Toshihiro OHIGASHI, Yuhei WATANABE, and Masakatu MORII, "Full Plaintext Recovery Attack on Broadcast RC4," Proc. the 20th International Workshop on Fast Software Encryption(FSE 2013), 2013.(Revised Selected Papers, LNCS 8424, pp.179-202, Springer-Verlag, 2014)。

*65 N.A.Fardan, D.J.Bernstein, K.G.Paterson, B.Poettering, J.C.Schuldt, On the Security of RC4 in TLS, USENIX Security 2013.(<http://dl.acm.org/citation.cfm?id=2534793>)。

*66 ImperialViolet, "POODLE attacks on SSLv3(14 Oct 2014)"(<https://www.imperialviolet.org/2014/10/14/poodle.html>)。

*67 Mozilla Security Blog, "The POODLE Attack and the End of SSL 3.0"(<https://blog.mozilla.org/security/2014/10/14/the-poodle-attack-and-the-end-of-ssl-3-0/>)。

*68 Google Online Security Blog, "This POODLE bites: exploiting the SSL 3.0 fallback"(<http://googleonlinesecurity.blogspot.jp/2014/10/this-poodle-bites-exploiting-ssl-30.html>)。

*69 「マイクロソフト セキュリティ アドバイザリ 3009008, SSL 3.0の脆弱性により、情報漏えいが起こる」(<https://technet.microsoft.com/ja-jp/library/security/3009008>)。

*70 SANS ISC InfoSec Handlers Diary Blog, "OpenSSL:SSLv3 POODLE Vulnerability Official Release"(<https://isc.sans.edu/diary/OpenSSL%3A+SSLv3+POODLE+Vulnerability+Official+Release/18827>)。

*71 mozillaZine, "Security.tls.version.""(<http://kb.mozillazine.org/Security.tls.version.>)。

効にする方法が共有されつつあります^{*72}。一方でレガシーな製品、特にフィーチャーフォンやゲーム機器などにおいては対策が難しいと考えられます。そのためSSL/TLSサーバがSSLv3非対応になることで、これらの機器からサイトが閲覧できなくなるケースが考えられます。

2) はダウングレード攻撃^{*73}を根本的に防ぐために提案されている方式です^{*74}。この方式ではTLS_FALLBACK_SCSVというCipherSuiteとエラーコードinappropriate_fallbackが新たに追加されており、クライアントとサーバの両者共にこのSCSV(Signaling Cipher Suite Value)^{*75}に対応している場合にのみ、ダウングレード攻撃を防ぐことができます。挙動は非常にシンプルでクライアントは自身が対応可能な最高バージョン以外でサーバに繋ぐときにはClientHelloメッセージのCipherSuitesに本SCSVを含むようにします。一方、本SCSVを含むCipherSuitesを受け取ったサーバは、クライアントが指定したプロトコルバージョンがサーバで対応している最も高いバージョンより低い場合にはエラーを返します。これらの挙動により潜在的なダウングレード攻撃を回避することができます。現在はTLS WGでLast callというステータスのドラフトですが、今年2月よりGoogleサーバとChrome(バージョン33以降)にて本SCSVが実装されています。また10月15日に公開されたOpenSSLでもこのSCSVが実装されました。更にFirefoxも対応することが予定されており^{*76}、今後サーバ・クライアント双方で利用が広がる見込みです。しかしこの対策はサーバ・クライアントの双方で対応しない限りPOODLE attackを防ぐことはできません。フィーチャーフォンなどのレガシーな環境において、この対策が浸透す

るとは考えられず、今後の潜在的な攻撃に備えて実装されている、と捉えることもできるでしょう。

POODLE attackの前提条件としてブラウザから大量にリクエストを発生させ、かつそのリクエストを経路上で一部書き換える(攻撃対象の暗号化データに差し替える)必要があることから、BEAST攻撃と同様に一般的な利用環境では容易に攻撃可能とは一概には言えません。しかし、今回はプロトコル仕様自体の問題であり、回避が難しいことからSSLv3はSSLv2と同様に脆弱なバージョンとして捉え、今後利用すべきではないと考えられます。実際TwitterなどのいくつかのサービスにおいてSSLv3無効化を行ったとのアナウンスが即座になされました^{*77*78}。

TLS仕様はTLSv1.0だけでなくBEAST攻撃に対処したTLSv1.1、より安全なハッシュ関数のSHA-256、SHA-384やCBC以外の暗号モードGCM、CCMに対応したTLSv1.2が規格化され、広く実装されています。更にTLSv1.3^{*79}も現在TLS WGにおいて検討されています。暗号モードとしてCBCを除外するなどの検討がなされており、最新のプロトコルバージョンを利用することで安全に利用できる仕組みが整いつつあります。かつては安全と認識され広く利用されていたDESやMD5などの暗号アルゴリズムが危殆化^{*80}し利用停止されたように、暗号プロトコルについても過去のバージョンを捨て速やかに新しいバージョンに移行していく必要があります。その際には、後方互換性(バージョンアップすることで繋がらなくなるSSLクライアントの比率)と安全性(対策の緊急度)のバランスを見ながら対応していくが必要になります。

*72 F5 Networks, "CVE-2014-3566: Removing SSLv3 from BIG-IP" (<https://devcentral.f5.com/articles/cve-2014-3566-removing-ssl3-from-big-ip>).

*73 意図的に古いプロトコルバージョンで接続させる中間者攻撃。ClientHelloにてクライアントが対応しているバージョンを明示する箇所があり、この部分をSSLv2であると指定することにより、意図せず弱い暗号アルゴリズムを利用させることができる。SSLv2ではこのメッセージはMACなどで守られていないためダウングレード攻撃の根本的な解決方法はなくSSLv2は脆弱なバージョンと認識されている。

*74 IETF, "TLS Fallback Signaling Cipher Suite Value (SCSV) for Preventing Protocol Downgrade Attacks" (<https://tools.ietf.org/html/draft-ietf-tls-downgrade-scsv>)。当初personalな提案だったドラフト (<https://tools.ietf.org/html/draft-bmoeller-tls-downgrade-scsv>) が現在TLS WG預かりに昇格していることが分かる。

*75 本来CipherSuiteはSSL/TLSで用いられる暗号アルゴリズムの組を示す2バイトデータであるが、2009年に発覚したRenegotiation機能の問題に対処するためにRFC5746 (<http://tools.ietf.org/html/rfc5746>) で導入されたTLS_EMPTY_RENEGOTIATION_INFO_SCSVと同様、本来とは異なる用途で別の意味を持たせる拡張が行われている。

*76 <http://www.ietf.org/mail-archive/web/tls/current/msg13905.html>

*77 <https://twitter.com/twittersecurity/status/522190947782643712>

*78 CloudFlare blog, "SSLv3 Support Disabled By Default Due to POODLE Vulnerability" (<https://blog.cloudflare.com/ssl3-support-disabled-by-default-due-to-vulnerability/>).

*79 The Transport Layer Security (TLS) Protocol Version 1.3 (<https://tools.ietf.org/html/draft-ietf-tls-tls13>)。ディスカッションは<https://github.com/tlswg/tls13-spec>を参照のこと。

*80 暗号危殆化の事例は本レポートのVol.8 (http://www.ijj.ad.jp/development/iir/pdf/iir_vol08.pdf) の「1.4.1 暗号アルゴリズムの2010年問題」にて紹介している。

1.4.3 リスト型攻撃の発生状況と対策

■ リスト型攻撃とは

昨年から様々なオンラインサービス(以下、サービス)において、リスト型攻撃またはリスト型アカウントハッキングと呼ばれる攻撃手法を用いた不正アクセスが頻発しています。リスト型攻撃とは、攻撃者がログインIDとパスワードを組み合わせたリストを用いて、サービスのログインを連続的に試行する攻撃です。攻撃に利用されるリストは、攻撃対象とは異なるサービスから流出した情報から作成されています。複数のサービスでログインIDとパスワードを使い回すユーザが多い^{*81}ことから、攻撃者はリストを利用して攻撃しているのです。

攻撃者がログインに成功しても、サービス提供者からは一見して、正規ユーザであるか否かを判別することは難しく、攻撃者は対策が行われる前に目的を達成することができます。ユーザがログインIDとパスワードの使い回しをしているサービスが多い程、ユーザの被害が大きくなってしまいます。

■ 他のログイン試行攻撃との違い

ログイン試行の攻撃手法としては、特定のログインIDに対して、様々なパスワードを試してみるブルートフォース攻撃や、パスワードとしてよく使われる単語やフレーズを試してみる辞書攻撃がよく知られています。リスト型攻撃では、試されるログインIDとパスワードの組み合わせが、あらかじめ決まっているという点がこれらの攻撃とは異なります。

■ 2014年のリスト型攻撃の発生状況

表-5は2014年1月1日から9月30日の間にリスト型攻撃として公表されたインシデントをまとめたものです。ほぼ毎月どこかのサービスがリスト型攻撃を受け、不正にログインされていることが分かります。金銭的被害も9件発生しており、クレジットカードで大量のチケットを購入し、換金する事例や、友人の名を騙ってメッセージを送り、ポイント

カードを購入させる事例、他にもサービス独自のポイントをギフト券などへ不正に交換する事例などが発生しています。リスト型攻撃の発見の契機については、サービス提供者による発見数13件、ユーザによる発見数9件となっています。サービス提供者自身もリスト型攻撃に気がついていない状況が多くあることが窺えます。

■ オンラインサービスにおけるリスト型攻撃対策

最後にサービス提供者側とユーザ側、それぞれのリスト型攻撃への対策を以下に説明します^{*82}。ただし、サービスを提供/利用する上での対策ですので、社内向けシステムなどには当てはまらない箇所もあります。

■ サービス提供者側のリスト型攻撃対策

サービス提供者側の主な対策は、(1)認証機能設計・実装の改善、(2)システム監視・運用の改善、(3)ユーザアクティビティ監視・通知機能の提供の3つです。なお、対策にあたって、ユーザに過度の負担を強くないような実装を考えなければなりません。

(1)の対策はログインID及びパスワード設定の改善やパスワード以外の認証機能の提供を行います。ログインIDとパスワードの使い回し防止のため、システムにポリシーに基づいたログインIDとパスワードを自動生成させることを検討してください。これにより、他サービスからのログインIDとパスワードの使い回しを防ぐことができます^{*83}。ユーザに自由なパスワード設定を許容する場合、「弱いパスワード^{*84}を受け付けない」「ログインIDとパスワードの使い回しをしないよう明示する」といった対策が必要となります。

その上で、使い回しをするユーザが必ずいることを想定し、多段階認証や多要素認証(ハードウェアトークンやICカード)で補強することを検討しなければなりません。この実装にはシステム改修が必要な場合があります。

*81 IPAとJPCERT/CCから公表された「STOP!! パスワード使い回し!! パスワードリスト攻撃による不正ログイン防止に向けた呼びかけ」(<https://www.jpccert.or.jp/pr/2014/pr140004.html>)の中で、金銭に関連したサービスサイトと同一のパスワードを使い回している人の割合が約4分の1(25.4%)であることが指摘されている。

*82 サービス提供者向けとして、総務省からも「リスト型アカウントハッキングによる不正ログインへの対応方策について」(http://www.soumu.go.jp/menu_news/s-news/01ryutsu03_02000063.html)が公表されている。

*83 サービス提供者が生成したログインIDとパスワードを、ユーザが他のサービスに利用することを技術的に防ぐことは困難であるため、ユーザに対し他のサービスに利用しないように提示する必要がある。

*84 「password」や「123456」、「ユーザの名前や住所、生年月日」など推測しやすいものや、それらの組み合わせは弱いパスワードとして設定できないようにしなければならない。

また、自社のシステムの脆弱性などにより情報が漏えいする可能性もないとは言いきれません。自社からログインIDやパスワードが漏えいした場合に備え、攻撃者が認証情報を手に入れても、リスト型攻撃に利用することが困難になるような対策が必要になります。パスワードを保存する際は、平文保存や単純なハッシュ化ではなく、ソルトを追加したパスワードにストレッチングを施したハッシュ化を行うことで、パスワードを現実的な時間で解析できなくなります。自社で実装することが困難な場合、外部の組織が提供する認証機能を利用することもできます。しかし、外部が提供する認証システムに対して攻撃が行われるといったリスクも考慮しなければなりません。

(2)は攻撃の事前・事後に備えた対策です。リスト型攻撃が行われた際、平常時と比べて、「システムの負荷が高い」「ログイン失敗の回数が多い」「存在しないログインIDに対するログイン試行が多数行われている」「特定のIPアドレスからのログイン試行が異常に多く、ログインIDが毎回異なる」「平常時とは異なるIPアドレスブロックからアクセスが行われている」といった兆候が見られる場合があります。

このような兆候を検知して、管理者への通知や攻撃元IPアドレス自動遮断、アカウントロックするようなシステムを検討してください^{*85}。また、長期間利用がないアカウントを削除することでリスト型攻撃による被害を未然に防ぐことができます。

表-5 2014年に発生したリスト型攻撃による事件

公表日	サービス分類	サービス提供者	期間	不正ログイン試行回数	不正ログイン件数	発見契機	金銭的被害	主な対策
1月24日	通信関連サービス	A社	1月16日	不明	165件	不明	なし	① ② ③ ⑤ ⑬
2月24日	ブログサービス	B社	不明	不明	不明	サービス提供者	不正なポイント交換	⑥ ⑩ ⑭
2月25日	SNS	C社	2月5日～2月11日	不明	370件	ユーザ	なし	① ⑥
2月28日	SNS	C社	2月28日	不明	16,972件	不明	なし	① ⑤ ⑦ ⑬ ⑭
2月28日	通信関連サービス	D社	2月24日～25日	約1万5,000回	344件	サービス提供者	不正なコンテンツ購入	② ③ ④ ⑤ ⑦ ⑩ ⑬
3月16日	ポイント関連サービス	E社	3月16日～3月17日	約94万回	約1万9,000件	サービス提供者	なし	④ ⑤ ⑦ ⑨ ⑬
4月23日	会員サイト	F社	3月23日～4月21日	460万回以上	78,361件	サービス提供者	なし	③ ⑤ ⑦ ⑧ ⑨ ⑬ ⑭
4月30日	通信関連サービス	D社	4月14日～4月28日	不明	724件	サービス提供者	不正なコンテンツ購入	② ③ ④ ⑤ ⑦ ⑩ ⑬
5月2日	ポイント関連サービス	G社	4月19日～4月29日	不明	273件	ユーザ	不正なポイント交換	② ④ ⑦ ⑨
5月30日	通販サービス	H社	2013年9月～2014年3月	不明	不明	不明	不正なコンテンツ購入	⑨
6月10日	動画サービス	I社	5月27日～6月4日	2,203,590回	219,926件	ユーザ	不正なポイント交換	⑬ ⑰
6月12日	SNS	J社	不明	不明	303件	ユーザ	ポイントカード購入	③ ⑥
6月17日	SNS	C社	5月30日～6月17日	約430万回	263,596件	ユーザ	なし	① ③ ⑥ ⑬ ⑮
6月20日	ブログサービス	B社	6月16日～6月19日	約160万回	2,398件	ユーザ	なし	⑫ ⑬
6月23日	ブログサービス	K社	6月19日～6月23日	2,293,543回	38,280件	サービス提供者	なし	⑬
6月26日	アンケートサービス	L社	6月23日～6月24日	不明	最大11,502件	ユーザ	不正なポイント交換	① ③ ⑰
6月30日	ゲーム関連	M社	6月28日～6月29日	1,796,629回	14,399件	サービス提供者	なし	③ ⑭
7月1日	広告サービス	N社	2013年8月ごろから不定期に発生	不明	不明	不明	不明	⑦
7月4日	アンケートサービス	O社	6月25日	3,420,000回	15,092件	サービス提供者	不正なポイント交換	③ ⑤ ⑬
7月4日	ゲーム関連	P社	不明	不明	不明	ユーザ	不明	⑤ ⑧
8月13日	通販サービス	Q社	8月7日～8月12日	4,220,382回	20,957件	不明	なし	⑬
8月18日	ポイント関連サービス	E社	8月15日	約296,000回	756件	サービス提供者	なし	⑬
8月27日	ゲーム関連	P社	不明	不明	不明	ユーザ	不明	⑤ ⑧ ⑪
9月11日	運輸サービス	R社	9月10日～9月11日	約1,152万回	約21,000件	サービス提供者	なし	⑤ ⑦ ⑧ ⑨ ⑬
9月23日	通販サービス	Q社	9月22日～9月23日	18,663回	19件	不明	なし	⑬
9月26日	運輸サービス	S社	9月25日～9月26日	約19万回	10,589件	サービス提供者	なし	① ⑤ ⑦ ⑧ ⑨ ⑬
9月29日	運輸サービス	T社	不明	不明	34,161件	サービス提供者	なし	① ⑤ ⑦ ⑧ ⑨ ⑬
9月30日	通信関連サービス	U社	9月27日～9月29日	約225万回	6,072件	サービス提供者	なし	① ⑤ ⑦ ⑧ ⑨ ⑪ ⑭

① 攻撃元IPアドレス遮断

② 検知システムの強化

③ 認証機能の強化

④ 監視体制の強化

⑤ 全ユーザにパスワード使い回し禁止を通知

⑥ 全ユーザにパスワード変更依頼

⑦ 全ユーザに定期的なパスワード変更依頼

⑧ 全ユーザに過去に使用したパスワード禁止を通知

⑨ 全ユーザに推測が容易なパスワード禁止を通知

⑩ 全ユーザにフィッシングサイトへの注意を通知

⑪ 全ユーザに2段階認証の利用推奨を通知

⑫ 不正ログインされたユーザを強制ログアウト

⑬ 不正ログインされたユーザをパスワードリセット

／変更依頼

⑭ 不正ログインされたユーザのアカウントをロック

⑮ 休眠アカウントをロック

⑯ システムの改修

⑰ 一部システムの停止

⑱ 不正な書き込みの削除

⑲ 警察へ捜査協力

*85 このような兆候は現在の攻撃傾向を反映した結果であるため、今後は傾向が変わる可能性がある。最新の攻撃傾向を情報収集し、それに合わせた観点で監視を行う必要がある。また、兆候の把握やそれを基にした自動遮断など、通信の秘密に対する適法性については、「電気通信事業者におけるサイバー攻撃への適正な対応の在り方に関する研究会 第一号とりまとめ」(http://www.soumu.go.jp/main_content/000283608.pdf)の「第4節 SMTP認証の情報を悪用したスパムメールへの対処」や「電気通信事業者における大量通信等への対処と通信の秘密に関するガイドライン」(http://www.jaipa.or.jp/other/mtcs/guideline_v3.pdf)の「(4) SMTP認証の情報を悪用した迷惑メールへの対処」などを参照のこと。

不正ログインが発覚した場合、被害の拡大防止と調査のため、サービスの停止を検討してください。不正ログインされたアカウントのパスワードは、再び不正ログインされないように、パスワードリセットなどで変更しなければなりません。また、念のため、今回、不正ログインの対象となっていないアカウントのパスワードを変更することも検討してください。

(3)は重要な処理の際にユーザにアラートを通知する機能です。ログイン成功・失敗時や個人情報の閲覧・変更時、購入決済処理などの完了時にメールなどでユーザに通知することで、ユーザ自身が不正アクセスに気付くことができます。また、ログイン履歴や購入履歴を確認できる機能を提供することで、いつ頃から不正アクセスが起きているのか把握することができます。

■ ユーザ側のリスト型攻撃対策

ユーザが自分でパスワードを設定する場合、パスワード管理ソフトウェアを使用して、サービスごとに異なるパスワードを生成・管理するのが最も効果的です。Webベースのサービスの場合、Webブラウザにパスワード生成ツールで作成したパスワードを記憶させる方法もあります^{*86}が、Webブラウザに起因するリスクに対応するため、マスター

パスワードは必ず設定し^{*87}、常に最新バージョンのWebブラウザを使用してください。

サービスが2段階認証を提供している場合は必ず使用するようになります。また、サービスの通知機能を有効にし、身に覚えのない利用通知がサービスから届いた場合は早急にパスワードを変更して、金銭的被害が発生していないか確認し、サービスの運営に連絡しましょう。また、使用する予定がなくなったサービスのアカウントは削除するようにしましょう。

1.5 おわりに

このレポートは、IIRが対応を行ったインシデントについてまとめたものです。今回は、Bashの脆弱性Shellshockについて、POODLE attack、リスト型攻撃の発生状況と対策についてまとめました。IIRでは、このレポートのようにインシデントとその対応について明らかにして公開していくことで、インターネット利用の危険な側面を伝えるように努力しています。このような側面についてご理解いただき、必要な対策を講じた上で、安全かつ安心してインターネットを利用できるように努力を継続して参ります。

執筆者:



齋藤 衛(さいとう まもる)

IJ サービスオペレーション本部 セキュリティ情報統括室 室長。法人向けセキュリティサービス開発などに従事後、2001年よりIIRグループの緊急対応チームIJ-SECTの代表として活動し、CSIRTの国際団体であるFIRSTに加盟。Telecom-ISAC Japan、日本シーサート協議会、日本セキュリティオペレーション事業者協議会など、複数の団体の運営委員を務める。

土屋 博英(1.2 インシデントサマリ)

土屋 博英、永尾 禎啓、鈴木 博志、梨和 久雄(1.3 インシデントサーベイ)

小林 直(1.4.1 Bashの脆弱性Shellshockについて)

須賀 祐治(1.4.2 POODLE attack)

小林 稔(1.4.3 リスト型攻撃の発生状況と対策)

IJ サービスオペレーション本部 セキュリティ情報統括室

協力:

加藤 雅彦、根岸 征史、春山 敬宏、桃井 康成 IJ サービスオペレーション本部 セキュリティ情報統括室

*86 Webブラウザで管理する場合、マルウェアに感染したときにパスワードが窃取される可能性があるが、多数の複雑なパスワードを身近なツールで管理できるという点において、現実的な手段の一つとして考えることができる。

*87 Firefoxはマスターパスワードを設定することができる。その他の主要なWebブラウザ(Internet Explorer、Google Chrome、Safari)では、OSなどのセキュリティ機構を使用してパスワードが自動的に暗号化されて保存される。

最新ストリーミング技術

トラフィックの増加が進むビデオストリーミングについて説明し、H.265、4K、MPEG-DASHについて解説します。また、夏の甲子園での実例についてご紹介します。

2.1 大規模配信

近年、ビデオストリーミングによるトラフィックは増加の一途を辿っています。IJJが提供しているストリーミングサービスでも毎年のように大容量の配信サイトを増設し続けています。Cisco社が発表しているVisual Networking Index(VNI)*¹でもビデオによるトラフィックの増大が予測されています。「ビデオトラフィックの総量は、2018年には全世界のコンシューマトラフィックの約80~90%になる」という見込みが発表されています。

このように、ビデオストリーミングは大規模化の傾向があります。実際、ちょっとしたイベントのライブ中継でも思ったより多くのアクセスを集めて驚くことがあります。ストリーミングのアクセス数予測は経験者でも難しい事柄ですが、「蓋を開けてみると思ったよりアクセス数が多かった」ということが増えています。

このようにストリーミングが大いに普及している理由は、モバイルデバイスの浸透にほかなりません。固定通信網(光

ファイバーやCATVなど)の契約数が伸び悩む一方、携帯電話の人口に対する普及率は2011年末には100%を越え、2014年6月では118%にまで達しています。またモバイルデバイス、特にスマートフォンに対する無線区間の提供速度が上昇してきたことも見逃せません。現在開発中の第5世代技術(5G)では、理想的な環境の場合にはダウンロードスピードが2010年比で100倍にもなるという試算結果が公表されています。将来、モバイルデバイスが生み出すトラフィックがこれまでの固定通信網のトラフィックを上回る可能性すら出てきます。

こうなると、コンテンツホルダーにとってモバイルデバイスは魅力あるセグメントに映ります。交通機関での移動中やほんのちょっとした待ち時間にモバイルデバイスを手にしている人を目にするが増えました。可処分である「空き時間」を巡って、多くのコンテンツホルダーが「モバイルへの配信」を検討し、そして実施しています。

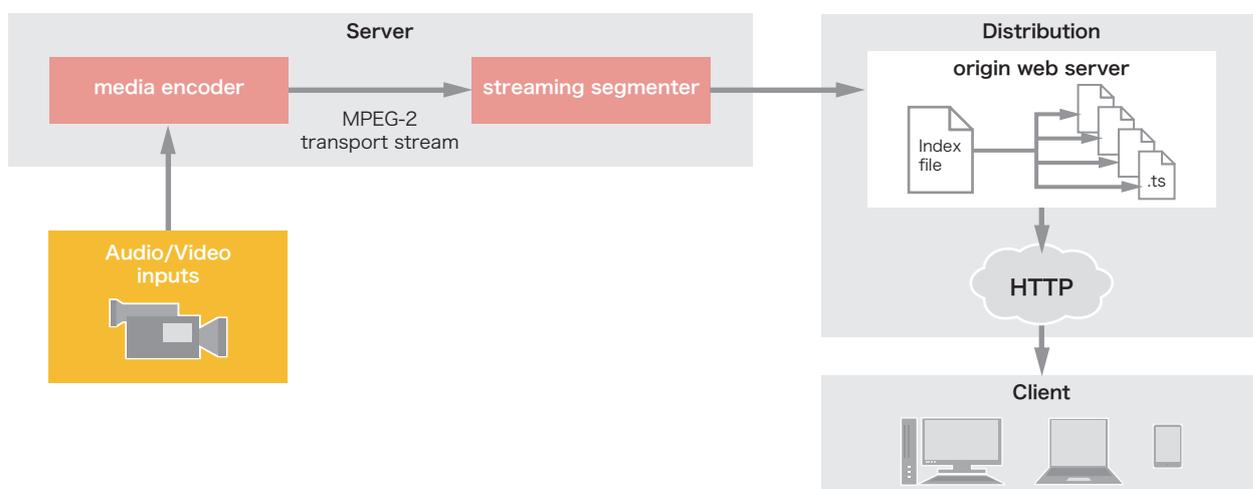


図-1 HTTP Live Streaming

*1 Cisco VNI(<http://www.cisco.com/web/JP/solution/isp/ipngn/connlife/vni.html>)。

2.2 モバイルの話

モバイルデバイスへの配信には、クリアすべき技術的なポイントがありました。その代表的なものが、デバイスがサポートしているストリーミングフォーマットです。OSでサポートしていないフォーマットを用いてクライアントへ配信しようとすると、そのストリーミング形式のライブラリをアプリに導入することが必要となります。これはライセンスコストや制作コストの上昇を招くため、手法として好まれません。したがって、モバイル用のOSがどのようなストリーミングフォーマットを標準で搭載するかが課題となってきました。

Appleは2009年にリリースしたiOS 3.0から、モバイルデバイスへのストリーミング配信をサポートしています。2009年という日本ではiPhone 3Gの販売開始から1年が経過し、iPhone 3GSがリリースされた頃です。HSDPAのサポートにより規格上は下り回線が7.2Mbpsとなり、モバイルへのストリーミングが現実化してきた時期に当たります。当時主流だったフィーチャーフォンでストリーミング受信が可能な機種はかなり限定されていました。iPhoneに対するストリーミング配信はWebサーバがあればよく、ストリーミングエンコーダの対応機種も増えたため、普及に時間はかかりませんでした。人々はスマートフォンで動画を見るという視聴体験を受け入れたのです。

Androidの場合、OSのバージョンによって状況が大きく変わります。Android 2.2や2.3がリリースされた頃はAdobe社からFlash PlayerがこれらのOS向けにも配布されており、動画配信はFlash技術(RTMP:Real Time Messaging Protocol)を用いる方法が主流でした。しかし2011年にAdobe社はモバイルデバイス向けFlash Playerの開発停止のニュースを発表。同時期にリリースされたAndroid 4.0でHLS(後述)がサポートされたため、その後はHLSによる配信が一般的となりました。

2.3 H.265の登場と4K

映像のデジタル処理は常にデータ量との戦いです。単純に図式化すると、画質を上げればデータ量は増加します。データ量を抑えるためには、画角(画面サイズ)を小さくするか、画質の劣化を受け入れるしかありません。映像を主観的に最小限の劣化に抑え、データ量を削減するための技術がCODECと呼ばれる技術です。CODECとは"CODER-DECODER"の略で、送出側でコーディングした(エンコーディングとも呼びます)ものを再生側でデコーディングすることを指します。この技術はDVDや地上デジタル放送で普及しています。現在、最も利用されているCODECは「H.264」*2と呼ばれる規格です。

H.264はITUとISOのジョイントグループで策定された規格です。策定されたのは2003年で、既に10年以上の歴史があります。この歴史の中でCPU能力の向上もあり、各メーカーではアルゴリズムの改良を進めてきました。H.264はあくまで規格であり、圧縮部分のプログラミングは各メーカーの独自の実装となるのです。こうした背景もありここ最近のエンコーダが作る映像は、初期のエンコーダに比べると画質がかなり向上しています。筆者は、H.264はまだこれから発展できるという研究者の意見も聞いたことがあります。しかし、そのH.264の性能向上を持ってしても、4Kへのキャッチアップは難しかったのです。

そこで登場したのが「H.265」*3という新しいCODECです。このH.265は開発時には「High Efficiency Video Coding (HEVC)」と呼ばれていました。H.265とHEVCとは同じものです。その名のとおり高効率で圧縮ができるもので、この10年間のCPU技術の向上を前提とし、動画圧縮に求められる計算量の増加を見越した規格となっています。またH.265はH.264の2倍の効率があると言われていました。2倍の効率とはつまり2倍のデータをH.264と同じ帯域で送る

*2 H.264: Advanced video coding for generic audiovisual service (<http://www.itu.int/rec/T-REC-H.264/e>).

*3 H.265: High efficiency video coding (<http://www.itu.int/rec/T-REC-H.265/e>).

ことができたり、H.264と同等の品質であってもファイルサイズは半分になるということを意味します。

現在、CODEC技術を持つメーカーは全力でH.265の性能向上に励んでいます。言い換えると、今後H.264 CODECに各社の更なる開発の手が加わることは考えにくいです(手を動かしているのは同じ開発者です)。結果として、H.264 CODECは現段階でかなり成熟しているといえることができるでしょう。

2.4 4K

4Kは、HD(High Definition)と比較してデータ量が非常に増えており、画角が4倍、フレームレートが2倍と、計8倍のデータ量があります。このままでは到底、放送や通信を用いて映像を配信することはできません。

限られた帯域しかない通信路においてデータが高い効率で圧縮されることは、より多くのユーザへの配信が可能となることを意味します。また帯域に限りがあるのは通信の世界だけではなく、放送においても同じことが言えます。多くのチャンネル数を確保するためには、1つのチャンネルが占有する帯域は少ない方が好ましいのです。日本においては4K放送でH.265の採用が決まり、現在試験放送が始まっています。また、今後次々と登場してくる通信を用いた4K映像サービスでも、H.265がCODECとして採用されています。どちらも「4Kを配信するためにはH.265の利用がマストである」と捉えられています。このように、H.265は4Kのインプラヤーとなりました。

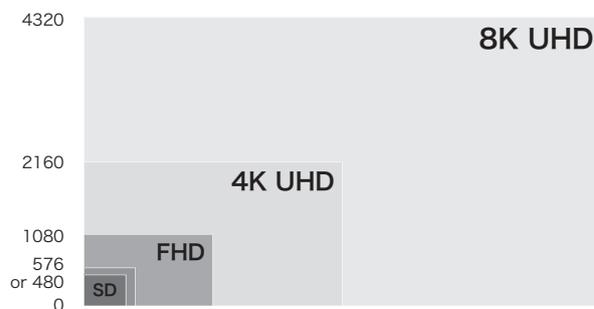


図-2 解像度の比較

2.5 MPEG-DASH

一方、インターネットでの配信方法は大きな変革期を迎えました。2000年代はIETFで標準化されたRTSP(Real Time Streaming Protocol)や、メーカー独自方式であるMMS(Microsoft Media Server)、RTMP(Real Time Messaging Protocol)が広く使われていました。これらはデータのトランスポートには基本的にUDPを使い、ファイアウォールを越す必要がある際にはHTTPでカプセルリングするということをしていました。しかし2010年代に入りストリーミングを直接HTTPを用いて配信する方法、つまりUDPを使わないプロトコルに移行してきました。

そもそも、HTTPでストリーミングを実装した先駆けはAppleが提案したHTTP Live Streaming(HLS)^{*4}です。Appleはこの規格を2009年以来IETFで提案していますが、ワーキンググループでの採択はされておらず、個人名義でのインターネットドラフト^{*5}が刊行され続けていますので、RFCにはなっていません。「ビデオデータを数秒に1度のセグメントに分割しWebサーバに配置。クライアントにはそのセグメントファイルを再生するためのURI(マニフェストと呼ばれる)をリストにして配布する」という非常にシンプルな構成で考えられています。実装はさほど難しくなく、多くのメーカーがこの方式に対応しました。一方HLSはインターネットの基準であるRFCになっておらず、規格がAppleのみによって定められるという状態が続いています。ストリーミングで必要とされているDRMもその規定はなく、ごく単純な暗号化しか定義されていません。全面的に依存すると、Appleの方針に従うほかなくなります。私見ですが、Appleは時折自分でルールを変える行動を起こします。そのリスクがHLSに波及しないかどうかを多くの会社が注視しているのではないのでしょうか。

例えば、Microsoftが提案したSmooth Streaming^{*6}もやはりHTTPストリーミングを採用していました。しかし、同社がMPEG-DASHを推進するようになって、新たに採用される数は減っていきました。Smooth StreamingはRich Internet Application(RIA)であるSilverlightの一部として

*4 HLS: HTTP Live Streaming(<https://developer.apple.com/streaming/>)。

*5 HTTP Live Streaming draft-pantos-http-live-streaming(<https://datatracker.ietf.org/doc/draft-pantos-http-live-streaming/>)。

*6 IIS Smooth Streaming Technical Overview(<http://www.microsoft.com/en-us/download/details.aspx?id=17678>)。

開発されていましたが、同アプリケーションの開発停止に伴いフォーマットとしての命運は尽きたと言えます。

Adobeが提案しているHTTP Dynamic Streaming (HDS)^{*7}にあっても、状況は同じです。特定の会社が標準化を踏まえ提案する技術を全面的に採用すると、ベンダーロックインのリスクが発生します。ましてストリーミング業界はここ数年でプロプラエタリなプロトコルからHTTPへ移行を開始したばかりです。自分ですべてが見渡せるクリーンな技術を求める空気がより強いかもしれません。

MPEG-DASH^{*8}はこのような状況に対するアンチテーゼとして生まれてきました。似たような技術であるにも関わら

ず、HLS、Smooth Streaming、HDSはそれぞれ異なったセグメント方法とマニフェストを採用しています。すると、再生クライアントやSTBは似たようなHTTPストリーミング方式であるにも関わらずそれぞれを実装するか、あるいは取捨選択を迫られてしまいます。このような状況を見て、ISO/IECのワーキンググループであるMPEG(Moving Picture Experts Group)は特定メーカーによるフォーマットによらない標準規格の策定に乗り出しました。2009年にRequest For Proposalが発行され、最終的に3GPP(Third Generation Partnership Project)が提案した方式が採用されました。MPEG-DASHとはDynamic Adaptive Streaming over HTTPの略です。ストリーミングプロトコルには標準化された規格であるHTTPを採用。更に、クライアントの受信状況

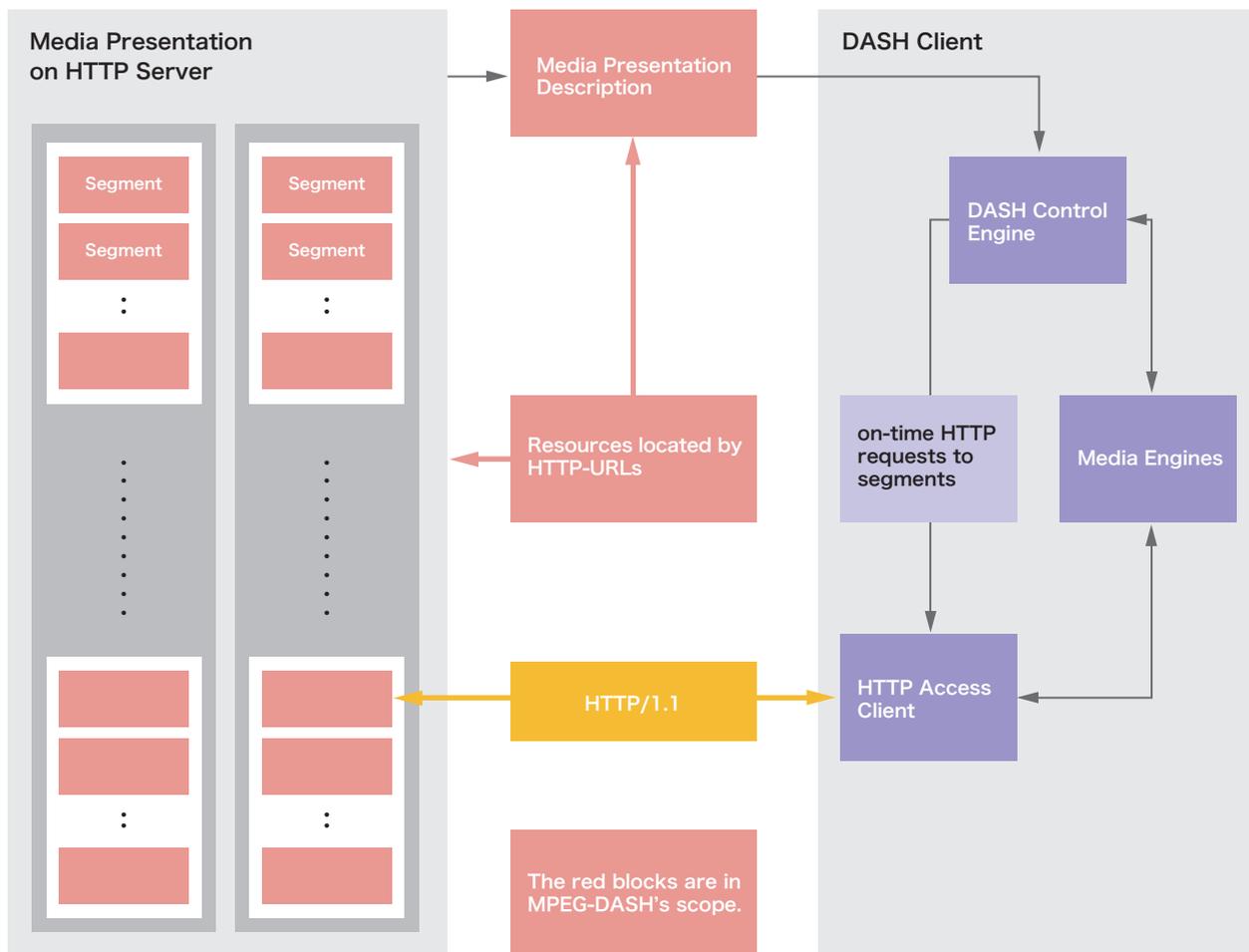


図-3 MPEG-DASH

*7 HDS: HTTP Dynamic Streaming (<http://www.adobe.com/jp/products/hds-dynamic-streaming.html>).

*8 MPEG-DASH ISO/IEC 23009-1:2014 Dynamic adaptive streaming over HTTP (DASH) -- Part 1: Media presentation description and segment formats: (http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=65274).

に応じてストリームを選択できるAdaptive Bitrateを採用しています。サーバ側で複数の帯域のストリームを用意しておくだけで、クライアント主導でストリームを切り替えることができるようになります。サーバ側での実装はまったく不要であり、クライアント側の総合的な環境変化に細やかに応じることができるため、Adaptive Bitrateは広く受け入れられる方式になっています。また複数のDRM方式の適用が可能であるなど、MPEG-DASHは今日の配信に求められる要求を幅広く備えています。

MPEG-DASHはまずヨーロッパで火が付き、今では世界的に採用が相次いでいます。普及促進を図るDASH Industry Forum^{*9}という団体もあり、NAB ShowやIBCといった放送業界向けの大規模カンファレンスでセミナーを開催するなど精力的な活動が目立ちます。日本ではIJJが他に先駆け2014年3月に実施された「東京・春・音楽祭」においてMPEG-DASHとH.265を用いたストリーミング配信を実施しました。これは最新技術をいち早く現場に応用した最新鋭の試みであり、複数のメーカ技術を組み合わせて実現

したものです。MPEG-DASHに対応したエンコーダやサーバの数は増えてきています。標準規格を選択することで多くのメーカが参入し、ユーザ側に選択の自由が生まれ、競争も促進されます。MPEG-DASHはマーケットを形成しつつあり、普及期に入ったといってもよいでしょう。

2.6 甲子園実例

朝日放送が取り組んでいる夏の甲子園のWeb展開において、IJJはストリーミング配信サービスを提供しています。朝日放送の特設Webサイトには例年大量のアクセスがあり、その多くがストリーミングのライブ配信を受信しています。これを混雑なしにストリーミング提供するため、IJJの大容量のCDNサービス(IJJ大規模配信サービス・マルチデバイスストリーミング/Live)をご採用いただきました。

2014年の取り組みでは、配信プロトコルはHLS(モバイルデバイス向け)とHDS(PC向け)を利用しました。これら二

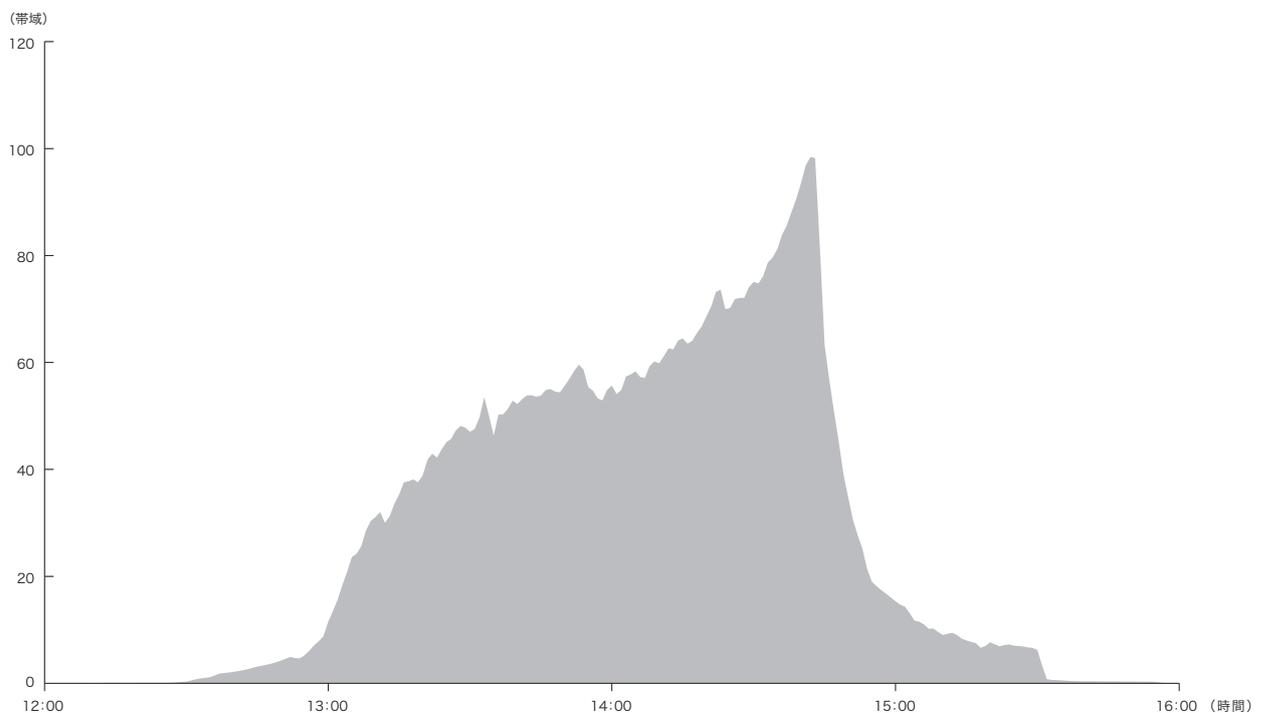


図-4 2014年8月25日 甲子園中継決勝戦のトラフィックグラフ

*9 DASHIF:DASH Industry Forum(<http://dashif.org/>)。

方式のプロトコルをひとつのプラットフォームから配信しています。朝日放送では映像をエンコーディングし、IJJのインジェストサーバへアップロードします。この段階で使われているプロトコルはRTMPです。IJJのサーバはクライアントからのリクエストを判断し、on-the-flyでHLS/HDSの出し分けをしています。お客様側でHLSとHDSの2つのプロトコルのストリームを用意する必要がなく、かつお客様手持ちのRTMP対応エンコーダを継続して用いることができるため、この機能は好評をいただいています。

今回一番注目を集めた技術は広告挿入です。野球のインニングの間に、朝日放送からのトリガーで広告ムービーを差し込むという事が行われました。今回は全ユーザに同じ広告を差し込みましたが、クライアントのIPアドレスやCookie、ユーザ認証などの技術と併用することで、広告のパーソナライゼーションが簡単に可能になります。冒頭に述べたように、今後のストリーミングビジネスではモバイルデバイスへの配信が大きなマーケットになります。モバイルデバイスは非常にパーソナルな存在ですので、そのような端末に対して広告挿入のカスタマイズができるということはビジネス上大きな可能性をもたらすと考えられています。

また、朝日放送としてモバイル向け甲子園中継は初の試みになりました。こうした試みを支えるのは、やはり安定した配

信です。配信がすぐに途切れたりすると、ユーザのストリーミング滞在時間は減少してしまいます。ビジネス上の機会損失に直結しますので、サービス提供者としては細心の注意を払いサーバの安定運用に努めなければなりません。

2.7 おわりに

ストリーミング技術はこのように標準規格の中で定められ、より大規模に発展していくものとなりました。HTTPはCDNへの適合性の向上やNAT/ファイアウォールフレンドリーになるなどのメリットをもたらしました。キャッシュサーバとの親和性も高いと考えられています。現在ではHTTP/2を用いたストリーミングを検討する動きもあります。一方、MPEG-DASHには国際標準をサポートする業界の支援（ツールやソリューションの提供）が受けやすくなるというメリットがあります。H.265も同様です。

IJJでは、今後もこうした最新技術のキャッチアップとサービス・ソリューションロールアウトを図っていきます。またIJJ技術研究所とのコラボレーションを進め、ストリーミングの配信をより大規模に、より安定させるための技術研究を進めていきます。

執筆者:



山本 文治(やまもと ぶんじ)

IJJ プロダクト本部プロダクト推進部企画業務課 シニアエンジニア。1995年にIJJメディアコミュニケーションズに入社。2005年よりIJJに勤務。主にストリーミング技術開発に従事。同技術を議論するStreams-JP Mailing Listを主催するなど、市場の発展に貢献。

SDNソフトウェアスイッチ「Lagopus」

SDNとOpenFlowの基礎から、最新のソフトウェアスイッチ「Lagopus」の性能にフォーカスして詳細を解説します。

3.1 SDNとOpenFlow

ネットワーク関連の業務に就かれている方であれば、SDNという言葉に耳にしたことはあるかと思います。ネットワークの構成要素として設置された、ルータやスイッチなど各種専用ハードウェア機器に人手で設定を施す従来の構築作業と比較して、「オーケストレータ」と呼ばれるシステムにより集中管理された情報を用い、ソフトウェアによってネットワークの構成要素に対して設定を投入する、それがSoftware Defined Networking (SDN)の基本的な考え方です。

SDNとOpenFlowを広く推進しているOpen Networking Foundation (ONF)によると、SDNの構成要素はおおよそ3つのレイヤーに分けることができるとしています(図-1)。

- アプリケーションレイヤー
- コントロールレイヤー
- インフラストラクチャレイヤー

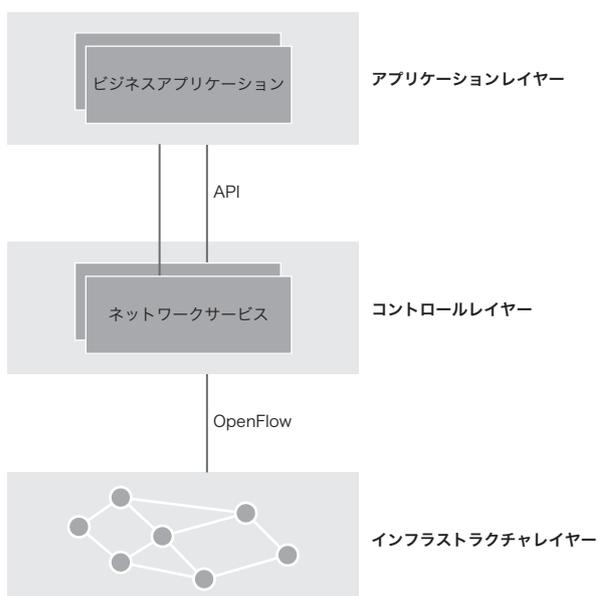


図-1 SDNの構成要素

アプリケーションレイヤーは、SDNの中核をなす集中管理とユーザインタフェースをつかさどるレイヤーで、ここに位置するのはSDNのみならずクラウド、すなわち仮想マシンについても管理を一手に引き受けます。ソフトウェアで言えばOpenStack、CloudStackといったオーケストレータが該当します。

コントロールレイヤーは、アプリケーションレイヤーとインフラストラクチャレイヤーの橋渡しの役目を担うレイヤーで、アプリケーションレイヤーから見るとネットワークサービスを提供するレイヤーとなります。アプリケーションレイヤーからの処理要求を、具体的な設定指示に置き換えて、インフラストラクチャレイヤーの各コンポーネントに設定を投入するのが主な役割です。ソフトウェアでいえばFloodlight、Ryuといったフレームワークを用います。

インフラストラクチャレイヤーは、文字どおりネットワークの基盤をつかさどるレイヤーで、顧客ネットワークの packets を送受信するコンポーネントが該当します。広義ではコントロールレイヤーからソフトウェア制御が可能なるルータ・スイッチの類となりますが、狭義では「ベンダーニュートラルなAPI・プロトコルによって」制御可能なものを指すことになります。

ベンダーニュートラルなプロトコルによって制御可能なスイッチの仕様として広く知られているものとして、OpenFlow Switch Specificationがあります。この仕様ではコントローラとスイッチ間の制御プロトコルも定義されていて、OpenFlow Protocolと呼ばれます。OpenFlowはオープンな標準として定義され、自由に仕様を参照してハードウェアやソフトウェアを実装することができます。OpenFlowに対応するインフラストラクチャレイヤーのコンポーネントとして、ハードウェアスイッチ製品の他にソフトウェアとして実装されたソフトウェアスイッチがいくつも存在しています。

3.2 OpenFlow

OpenFlowは2008年にスタンフォード大学で策定され、その後仕様の拡張・修正がなされ、現在はONFによって定義されているオープンな技術仕様です。仕様にはバージョンが振られていて、現在最も多く使われているのはOpenFlow 1.0です。その後いくつかの拡張がなされ、1.1や1.2が策定されましたが、普及前にバージョン更新されたためにあまり使われず、長期サポートが謳われているOpenFlow 1.3の実装がよく見られます。執筆時点での最新バージョンはOpenFlow 1.4ですが、仕様の複雑化による実装の困難さから、対応した実装はほとんどないというのが現状です。OpenFlowでは、バージョン間の互換性を考慮しない仕様となっているために、コントローラとスイッチの間で利用するバージョンが一致していなければなりません。

OpenFlowの考え方は、制御プレーンとデータプレーンを分離するところから始まっています。データプレーンは、与えられたルールに従ってパケットを高速に処理・転送する役割、制御プレーンはルールを与える役割になります。この場合のルールとは、受信したパケットがどのようなものかを判別するものであり、OpenFlowでは1つのルールとそれに対応した処理を一組として、フローエントリと呼んでいます。例えば、「LANポート1で受信した、宛先IPアドレス192.168.0.1の、TCP80番ポート宛のパケット」といった

ルールを与えることができます。複数のフローエントリをまとめたものをフローテーブルと呼び、OpenFlowで処理されるパケットはフローテーブルの中から該当するフローエントリ1つを見つけ出し、関連付けられているアクションを実行することになります。アクションには、単純に指定されたLANポートへパケットを送信するといったもののほか、宛先IPアドレスの書き換え、VLANタグの挿入といったパケットヘッダの変更などもあり、これらは仕様に定められた範囲で自由に指定することが可能です。フローテーブルを複数用意しておき、アクションとして次のフローテーブルの処理をせよ、と指定することもできます。

また、アクションにはパケットを制御プレーンつまりコントローラに送信するという指定が可能です。コントローラはパケットをスイッチから受け取り、より複雑な処理を実行することができます。更に、コントローラはスイッチに対して、任意のパケットを送出させることもできます。これらをまとめると図-2のようになります。

このように、OpenFlowではコントローラとスイッチの連携によって、高速性を保ったまま複雑なパケット処理を実現することを考慮して考案されました。しかし、OpenFlow仕様自体の拡張によって、データプレーンは複雑化の一途をたどっています。

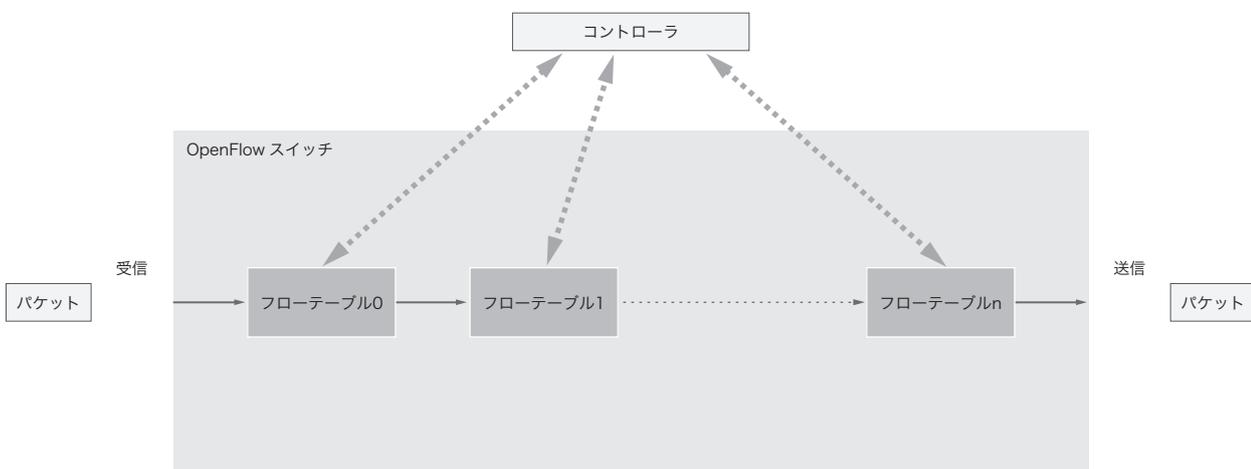


図-2 OpenFlowの考え方

3.3 ソフトウェアスイッチ

OpenFlowに対応するEthernetスイッチの実現方式としては、大まかに3つに分類できると言えます。

1つ目は、ASICを用いたハードウェアスイッチ製品です。スイッチ容量は大きく、大規模ネットワークに対応できる性能を持つ半面、扱えるフローエントリ数やテーブル数に相当な制限がかかり、かつ価格も高価です。また、OpenFlow仕様の項目には必須なものとオプションとして省いていいものがありますが、ハードウェアスイッチ製品では実装の難しい処理が省かれていることが多い点に注意が必要です。

2つ目は、ネットワークプロセッサ(NPU)を用いた製品です。ネットワーク処理のための特殊な機能を内蔵したプロセッサを利用し、その機能を活かすファームウェアを内蔵したものです。ASICを用いたスイッチと比較すると、フローエントリ数、テーブル数、実装項目共に制限は緩く、また速度もハードウェアスイッチには及ばないことがあるものの、大規模ネットワークにも適用可能と言えます。やはり価格は高価であり、中規模あるいは小規模ネットワークに適用するにはコストが見合いません。

3つ目は、ソフトウェアスイッチです。サーバの汎用OS(例えばLinux)で動作するプログラムによってOpenFlowの機能を実現します。仮想マシン上で動作させることから、クラウドとの相性がいいという利用面の性質があります。機能的には実装次第でOpenFlowの全機能を提供可能な半面、性能についてはASICやNPUの足元にも及ばないというのがこれまでの現状であり大きな課題でした。

10GbEネットワークインタフェースで64バイトのパケットをワイヤーレートで処理すると、14.88Mフレーム/秒(fps)となり、1フレームあたりにかけることのできる時間は67ナノ秒という計算になります。プロセッサクロックが2GHzとすると、1クロックは0.5ナノ秒になります。大雑把にい

えば、1クロック1命令と仮定した場合、134命令で1パケットの処理を完了させなければワイヤーレートでのパケット転送は実現できないということになります。2012年当時、既存のソフトウェアスイッチ実装ではせいぜい数百K~1Mフレーム/秒であったことから、性能面でのボトルネックを解消すべく、新しいソフトウェアスイッチの実装が検討されました。そうして開発されたのがSDNソフトウェアスイッチLagopus(ラゴパス)です。

3.4 SDNソフトウェアスイッチLagopus

Lagopusは、OpenFlow 1.3仕様に準拠したソフトウェアスイッチです。2013年に開発が始まり、2014年7月にApache Licenseでソースコードが公開されました。2014年10月現在の最新バージョンは0.1.1で、現在も開発が続いています。Lagopusの主な特徴を列挙すると下記のとおりになります。

- 64 Linux用ソフトウェアスイッチ
- 1プロセス、マルチスレッド動作(2コア以上必要)
- 開発言語は主にC言語
- OpenFlow 1.3.4準拠
- OpenFlow仕様に記述されているオプションをほとんど実装
- MPLS - VLAN変換を伴うパケット転送で10GbEワイヤーレートの性能
- フローエントリ数:100万、テーブル数:254の大規模設定に対応
- Intel DPDKを利用したユーザスペースによる高速パケット処理
- Apache Licenseによりソースコード公開、非商用・商用を問わず利用可能なオープンソースソフトウェア

本稿では、これらの特徴のうち性能にフォーカスして、詳細を解説します。まず、ソフトウェアスイッチにおいて性能に大きく影響の出る主なポイントについて書き出してみます。

3.5 ソフトウェアスイッチのボトルネック

3.5.1 受信割り込み、送信完了割り込み処理

CPUが様々な処理を実行している最中に、ネットワークインタフェースカード(NIC)がパケットの受信を検知すると、NICはパケットのバイト列をバッファメモリに格納した上で、CPUに対して割り込みを発行します。CPUはNICからの割り込みを検知すると、処理中の状況(コンテキスト)を一時的に退避して、割り込み処理に移ります。割り込み処理では、バッファメモリに格納されたパケットの情報を取り込み、ネットワークスタックの受信キューに詰めるなど最低限の処理が実行され、処理が完了すると退避していたコンテキストを復帰させ、割り込みがかかる前の処理を続行します。このコンテキストの切り替わりのことをコンテキストスイッチと呼ぶのですが、高速なパケット転送をする上ではこれが大きな負荷となります。1パケットずつ割り込みを発生させていたのでは性能に著しい影響があるため、NICの機能としてある程度まとまったタイミングで割り込みを発生させるといった負荷を軽減する工夫があります。

3.5.2 パケットのメモリコピー

また、受信したパケットはまずOSのカーネル空間で管理されるメモリに記録されますが、通常これを直接ユーザスペースのプログラムから読み書きすることはできません。システムコールread(2)あるいはrecv(2)は、ユーザプログラム側であらかじめ確保したメモリ空間に、受信したパケットの内容をコピーします。逆に、システムコールwrite(2)あるいはsend(2)は、ユーザプログラムが用意した内容をカーネル内のメモリ空間にコピーします。10GbEワイヤレートであれば1秒間に14.88Mフレームを送受信することになりますので、64バイトのパケットであれば秒間 $14.88M \times 64 \times 2 = 1.9GB$ のメモリコピーが発生することになります。DDR3-1600のメモリ転送速度の理論値は12.8GB/秒ですので、単純に当てはめたとしてもメモリコピーだけで全体時間の約15%を消費することになります。

3.5.3 OSによるプロセススケジューリング

OSのユーザスペースで動作しているプロセスは複数ありますが、psコマンドなどで見れば分かるとおりマルチコア全盛の現在においても搭載されているCPUのコア数で賄える数にとどまることはありません。そのため、OSは適宜

動作させるプロセスを切り替えて、見た目上同時に動いているように見せています。ミリ秒単位で切り替わっていくため通常人間が触っている限りでは気付かないほど高速ではあるのですが、ソフトウェアスイッチの動作中にこのようなOSによるプロセススケジューリングが発生すれば、当然ながら性能に大きく影響します。

3.5.4 メモリブロックの確保と解放、ページフォルト

読み書きをする際に使用するメモリブロックについては、受信のたびに確保し、送信完了のたびに解放するのは効率的ではなく、すぐにまた次のパケットを受信するということを考えると領域を再利用することにより性能の低下を軽減できます。しかし、ユーザスペースにあるメモリはすべてが実際のメモリ上に置かれているとは限りません。OSは必要に応じてユーザスペースのメモリと実際のメモリをマッピングします。必要に応じてというのは、プログラムがそのメモリにアクセスしようとした瞬間です。この仕組みをオンデマンドページングと呼びますが、処理としてはプログラムがメモリにアクセスしようとした瞬間に例外が発生し、例外処理の中でメモリのマッピングを実行し、復帰後プログラムは何事もなかったかのように処理を実行し続けます。この例外をページフォルトと呼びます。ページフォルトが発生してメモリのマッピング処理が実行される場合と、既にメモリがマッピング済みである場合とで、処理時間に大きな差があることは自明です。

3.5.5 リソースへの排他アクセス

OpenFlowではフローテーブルを参照してパケット処理を実行することは説明しましたが、当然ながらパケット処理中にもフローテーブルへのエントリの追加・変更・削除が可能ですが、受信したパケットにマッチし今まさに参照しているフローエントリが削除されることがあってはなりません。そのため、パケット処理とフローテーブルへのエントリ追加・変更・削除は互いに排他的に実行できなければなりません。排他アクセスが必要なリソースはフローテーブルのみではなく、例えばLANポートのカウント情報などがあります。通常このような用途にはリーダー・ライターロックと呼ばれる仕組みを使うのですが、ロックは比較的重い処理で、1パケットごとにロック・アンロックを繰り返しては転送性能に大きな影響が出ます。

3.6 ボトルネックを解消するIntel DPDK

前出のボトルネックのうちのいくつかを解消しようという試みは、既にいくつかありますが、ここではLagopusで利用しているIntel DPDKの取り組みを紹介します。Intel DPDK (Data Plane Development Kit)は、データプレーンをプログラミングするにあたって有用なAPIを提供するライブラリであり、高速なパケットI/Oを実現するフレームワークとなっています。執筆時点での最新バージョンは1.7.1です。Intel DPDKの大きな特徴は下記のようなものです。

- x86、x86_64 Linux及びFreeBSDで動作
- BSD Licenseで提供されているため、商用・非商用を問わず利用可能
- 割り込みを用いないパケット送受信を実現するPoll Mode Driver (PMD)
- ユーザスペースI/Oによるゼロコピー
- マルチコア前提、コア固定によるOSのスケジューリング回避
- hugepageを利用したページフォルトの削減
- Longest Prefix Match、hash、リングバッファなど最適化されたライブラリ
- サンプルアプリケーションを多数同梱

3.7 Lagopusにおける性能向上策

Lagopusでは、Intel DPDKを利用することで、性能ボトルネックの多くを解消しています。とりわけ、PMDとコア固定は大きく性能向上に貢献しています。また、Lagopusはスレッド構成が特徴的と言えます(図-3)。

Lagopusのデータプレーンは、パケットI/Oを担当するスレッドとOpenFlow処理を担当するスレッドが分かれています。更に、I/Oは受信(RX)と送信(TX)に、OpenFlow処理(worker)は複数のスレッドとなるよう起動時に指定が可能です。これらのスレッドは、それぞれ他の処理が割り当てられないよう別々のコア上で動くよう固定させることで、性能を確保しています。また、スレッド間の通信には、DPDKが提供しているロック処理不要のリングバッファを用いています。

パケットI/Oの基盤部分は主にIntel DPDKを利用することでボトルネックを解消していますが、OpenFlow処理部分ではLagopus独自の性能向上策が見られます。Lagopusのフローテーブルの内部構造を図-4に示します。

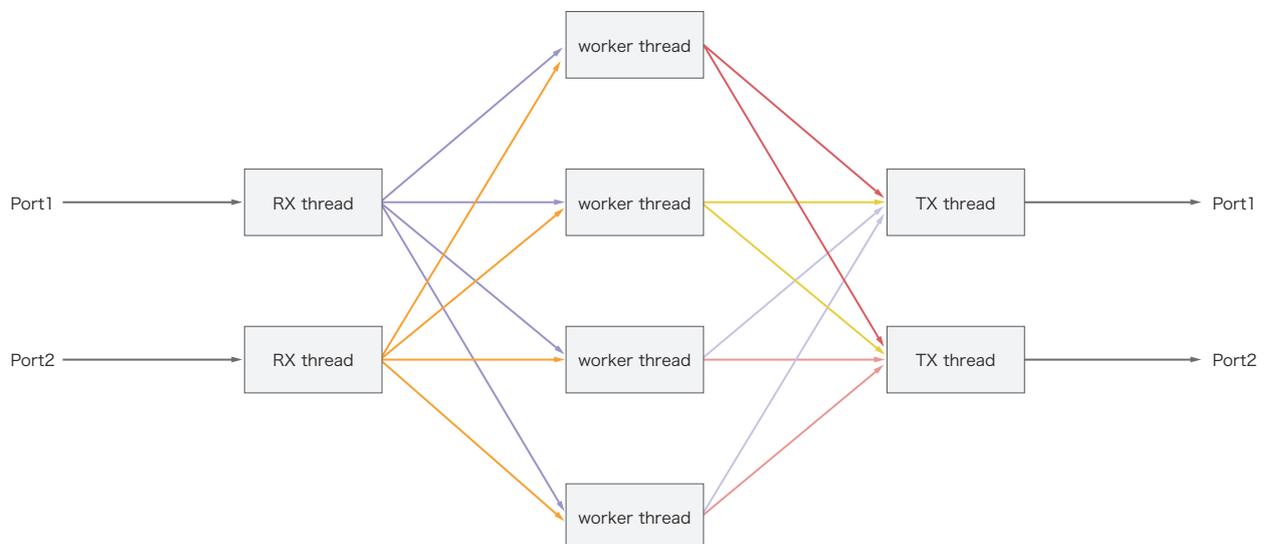


図-3 Lagopusのスレッド構成

コントローラからの要求に応じてフローテーブル内容を出力する必要があるため、OpenFlow仕様を素直に表現したフラットなテーブルが用意されていますが、Lagopusではそれとは別にフロー探索に適した構造のテーブルが用意されています。この構造は現在のところ固定的で、どんなフローテーブルでも最適な探索ができるというものではありませんが、VLAN IDによる分岐、MPLSラベルによる分岐、IPv4宛先アドレスによる分岐、といったように、典型的に使われるであろうパケットヘッダに対して高速な探索を実現しています。また、一度探索したパケットの情報をキャッシュし、同じパケットを受信した際には、キャッシュからフローエントリ情報を取得し探索自体を省略するという、フローキャッシュの仕組みも実装されています。

以上の性能向上策の相乗効果により、Intel Xeonを用いた試験環境において、10万エントリのフローテーブルを投入した状態で、OpenFlowによるパケットヘッダ書き換えを

伴うパケット転送にて、Lagopusは10GbEワイヤードレート(14.88Mfps)を達成することができました。そして現在Lagopusは、より複雑なフローエントリでの処理速度向上や、より多くのポートを同時に処理する際の性能の維持、更に、より高速なインタフェースへの対応を順次進めています。

3.8 まとめ

SDNソフトウェアスイッチLagopusは、OpenFlow仕様への準拠度も高く、従来のソフトウェアスイッチと比較して飛躍的な性能向上を実現しています。現在Lagopusは試験環境以外でのテストや動作確認が不十分なプロトタイプ品質のソフトウェアですが、実環境にて幅広く使用されることを想定して開発が続いています。もしご興味がありましたら是非ホームページ(<http://lagopus.github.io>)をご覧ください。

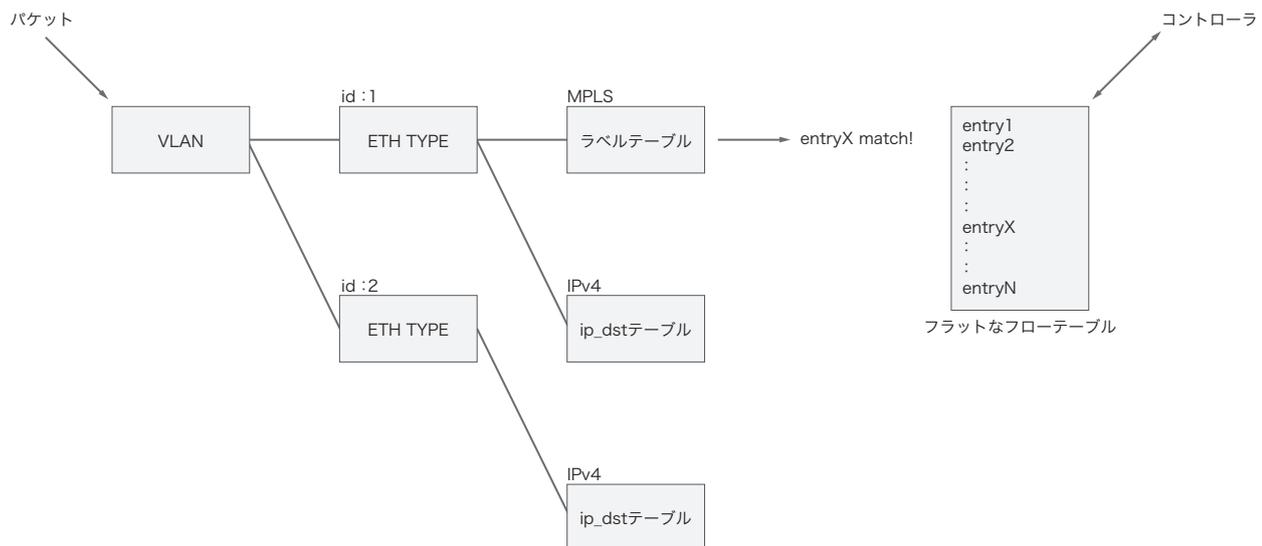


図-4 Lagopusのフローテーブルの内部構造

執筆者:



沖勝(おきまさる)
株式会社ストラトスフィア 研究開発部。

株式会社インターネットイニシアティブ(IIJ)について

IJは、1992年、インターネットの研究開発活動に関わっていた技術者が中心となり、日本でインターネットを本格的に普及させようという構想を持って設立されました。

現在は、国内最大級のインターネットバックボーンを運用し、インターネットの基盤を担うと共に、官公庁や金融機関をはじめとしたハイエンドのビジネスユーザに、インターネット接続やシステムインテグレーション、アウトソーシングサービス等、高品質なシステム環境をトータルに提供しています。

また、サービス開発やインターネットバックボーンの運用を通して蓄積した知見を積極的に発信し、社会基盤としてのインターネットの発展に尽力しています。

株式会社インターネットイニシアティブ

〒102-0071 東京都千代田区富士見2-10-2 飯田橋グラン・ブルーム
E-mail: info@ij.ad.jp URL: <http://www.ij.ad.jp/>

本書の著作権は、当社に帰属し、日本の著作権法及び国際条約により保護されています。本書の一部あるいは全部について、著作権者からの許諾を得ずに、いかなる方法においても無断で複製、翻案、公衆送信等することは禁じられています。当社は、本書の内容につき細心の注意を払っていますが、本書に記載されている情報の正確性、有用性につき保証するものではありません。

©2008-2014 Internet Initiative Japan Inc. All rights reserved.

IJ-MKTG019YA-1411GR-09200PR