TCP Vulnerability 対策について

2004年 7月7日（七夕） 吉田 友哉（たなばた）
Yoshida Tomoya / yoshida@ocn.ad.jp
NTTコミュニケーションズ

1E7D 79AD C610 B5F2 A94E 7FF4 F4AC A722 329C 3DE8

本日のアジェンダ

- TCP Vulnerability のおさらい
- MD5関連の諸々
- 対応策について
- ディスカッション

TCP Vulnerability アナウンス

- 実際には、4/21 日本時間明朝にアナウンス
  - 本当は21日の夜21時だったが、マスコミが先走ってしまったらしい (真相は、JPCERT様、御願いします)
  - NISCC Vulnerability Advisory 236929
    - http://www.uniras.gov.uk/vuls/2004/236929/
  - この前後に、ISP間でMD5が盛んに実施される

- JPCERTより、4/19に優先情報提供
  - 「4/21夜21時に何かが発表される」
  - BGP MD5を実装し、安心して21日の夜を迎えよう！
    - 実際にはどういったSecurity Hole なのか？
    - 本当にMD5を実施すればよいのか？
    - 他にも方法があるのではないか？　など様々な意見

満を持して出てきたドラフト

- 4/21 draft-ietf-tcpm-tcpsecure-00.txt
  - 主にRSTやSYNに対する実装の改良に関するもの
  - RFC793のadd-on
  - 事前に実装済みコードが準備できたベンダ vs draftが出てから実装しはじめたベンダ
    - 主なルータベンダ vs L3SWベンダ　の構図？
  - 現在は draft-ietf-tcpm-tcpsecure-01.txt

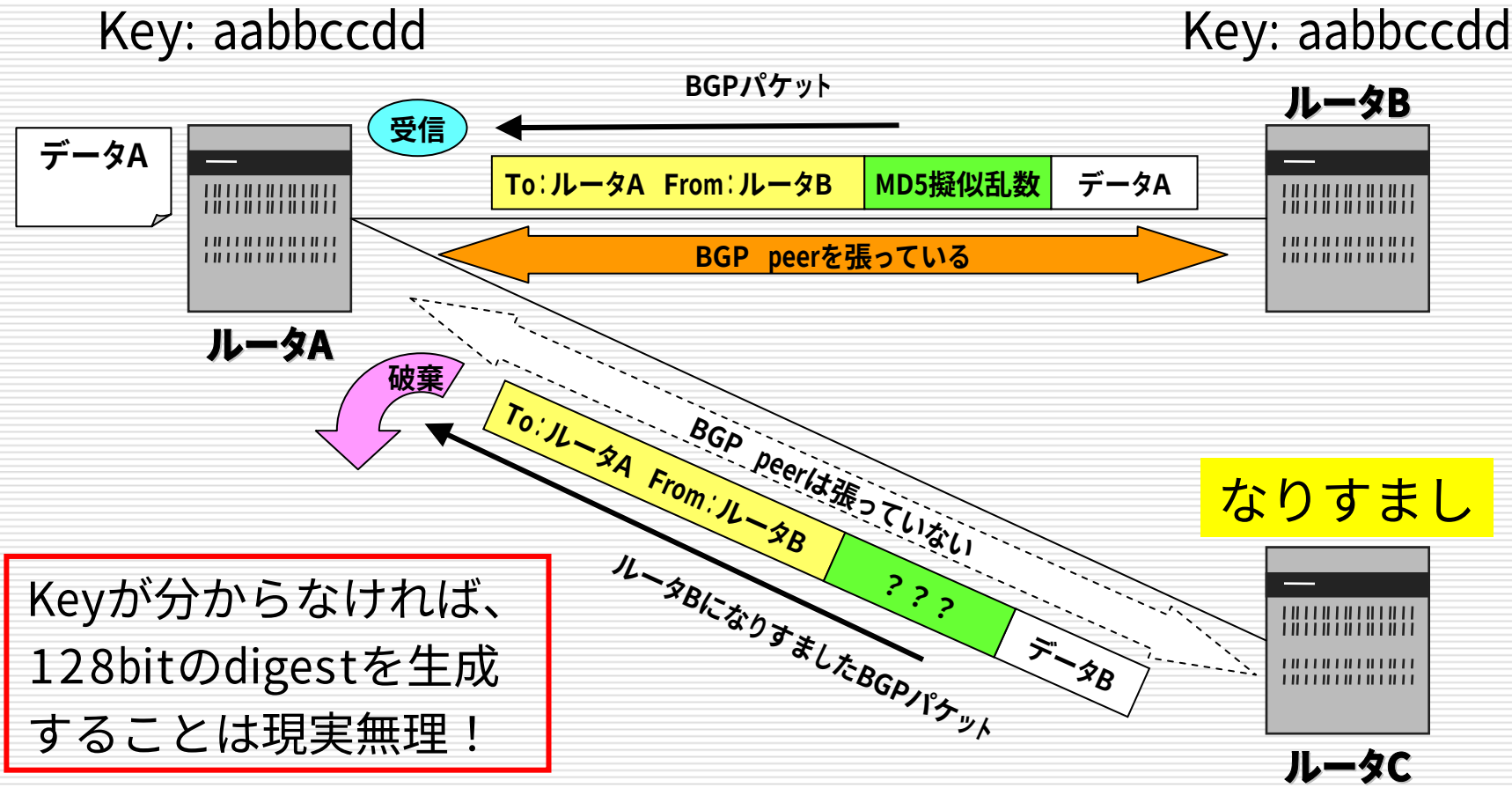
TCP Vulnerability 何が痛い？

- BGPのような長いTCPセッションが痛い
  - Port番号が長いこと変わらない
  - 他には、DNSのゾーン転送のセッションも痛い？
- 特にeBGPが痛いのは明白
  - アドレスがばればれで、BGPの片方は179番Port
- 意外にPort番号の推測が容易で、ある程度推測された形でのAttackがなされる可能性がある
  - 某社のeBGPセッションは、11000 Port から使う（プラットフォームやOSによっては異なる模様）
  - Window size もけっこう大きい
    - Window size ごしにやられると、ものの数分でsequence number を1周出来る（だいたい20万強回で可能）

MD5とは

- Message Digest 5
  - 認証やデジタル署名などに使われるハッシュ関数(一方向要約関数)のひとつ、認証アルゴリズム
  - 両端で同一なキーを設定し、MD5アルゴリズムを用いて変換された128bitの固定長のbit列を両端で比較することで、改ざんされていないか確認
  - MD2、MD4 → MD5
  - 簡潔さ，安全性，速度を重視
    - SHA-1(Secure Hash Algorithm) : 160bit
  - RFC1321

MD5認証イメージ



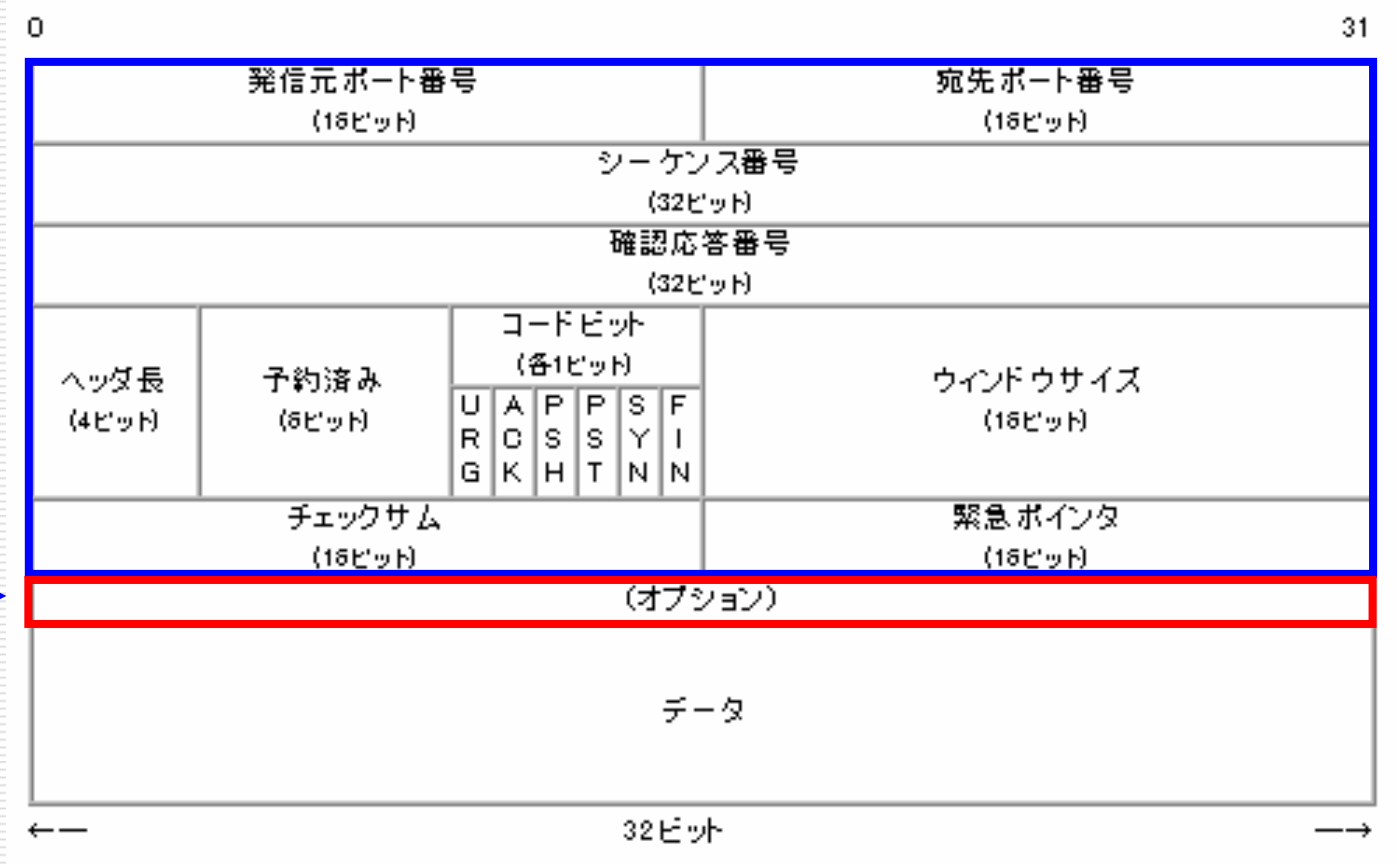
Keyが分からなければ、128bitのdigestを生成することは現実無理！

BGP MD5 Algorithm (RFC2385) 抜粋

Every segment sent on a TCP connection to be protected against spoofing will contain the 16-byte MD5 digest produced by applying The MD5 algorithm to these items in the following order:

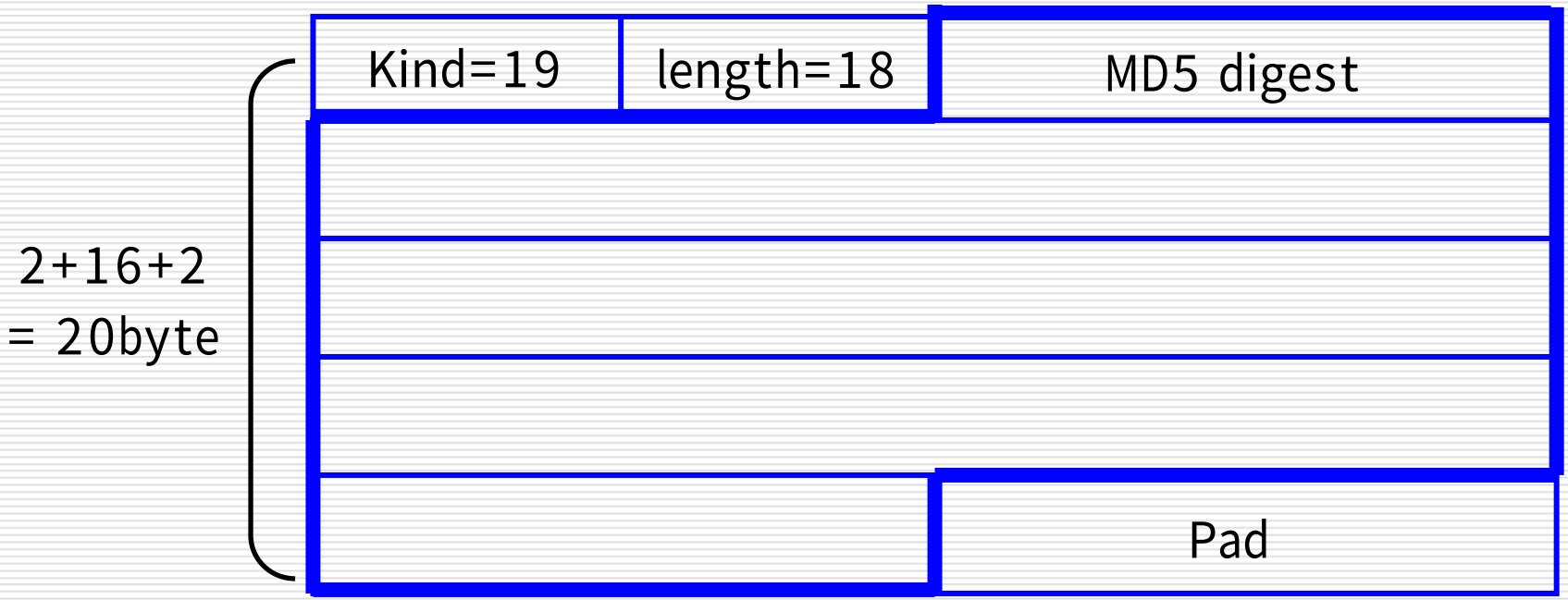
1. the TCP pseudo-header (in the order: source IP address, destination IP address, zero-padded protocol number, and segment length)
2. the TCP header, excluding options, and assuming a checksum of zero
3. the TCP segment data (if any)
4. an independently-specified key or password, known to both TCPs and presumably connection-specific

TCPヘッダ (参考)



MD5オプション

RFC2385で定義



MD5を使ってみると・・・

- 一部を除き、多くのベンダで実装済
- しかし、今まで使われてきていなかったという事実が浮き彫りに
  - 細かい部分の実装や関連部分に差分あり
    - ベンダによっては、digest key の文字制限や解釈の差分がある
  - 間に別の装置が入っているケースなど
- Keyの交換、管理方法が非常に大変
  - さすがにメールでのやり取りを拒み、電話が多い
  - PGPでやろうとする人は少ない
- ISPをまたがった相互接続性の問題も
  - 相手が何の装置を使っているかを何気なく聞く？
  - Macから調べてみる？（最近はOEMなんかで判別困難）

MD5な諸々

- MD5を使われることを想定していない装置
  - MD5ハッシュ値は、TCPヘッダを元に生成
    - ヘッダの中身変更 ⇒ 当然ハッシュ値変化
    - 可能性としては、window, flag, etc...
    - ヘッダ内容を変更する or 省くなどの装置が介在すると、セッション張れない
    - L2動作で替えられるとそもそも装置の存在に気付きにくく、おまけにキャプチャしても一見正常な場合も
    - ハッシュ値不一致を示すログがあれば分かり易い

MD5な諸々 (Cont.)

- MD5でTCPヘッダサイズは拡大(40Bytes)
- MSSの扱い次第では、パケットがフラグメント
  - RFC2385／通知MSS減算
    - 相手に通知するMSSのサイズは、あらかじめTCPのオプションを考慮した上で減らして通知すべき
  - RFC879／実際送信するMSSの選択
    - パケットを送信する際に自分と相手側のMSS値を比較し、小さい値を選ぶ＝Optionを考慮しなくても良いとの記述

MD5な諸々 (Cont.)

- MSS (Maximum Segment Size)
  - 3Wayの時にやりとりされる
  - MD5無し→option無し
    MSS=1500-(IP20+TCP20)=1460
  - MD5有り→option有り
    MSS=1500-
    (IP20+TCP(20+18+2))=1440

MD5な諸々 (Cont.)

- 相性次第では、フラグメント発生
  - → 自分が通知したMSSで相手がそのまま送ってくる
  - → 通知時に減算しないと、フラグメントする
- 通常フラグメントしても問題ない

主なIXでの対応

- dix-ie
  - 基本はISPにおまかせ（4月の ixp-meetingにて、事前に呼びかけ実施）
- JPIX
  - IXの関与の仕方があるかもしれない
  - Publicな情報をもとに、MLにて対応を促す
    - → IXのセグメントに対する一部対処
- JPNAP
  - 基本はISPにおまかせ
  - Publicな情報をもとに、MLにて対応を促す

MD5以外の対処方法は？

- TTL Hack
- RPF Check
- Filtering
- Using Private Address
- BGP No Announcement
- DNS (おまけ)

TTL Hack

- 設定したTTL値の範囲外のTTL値をもったパケットが来た場合には、はじいてしまう
- BTSH/RFC3682 (GTSM)
  - The Generalized TTL Security Mechanism
- eBGP部分では非常に有効。ただISP間で動作させるには、互いに仕様などの確認が必要だろう
  - 対向機器が変更になった場合などの対応も出てくるだろう
- 現状ベンダにてほとんどimpleされていない
- iBGPに適応するのは難しいかも
  - TTL値が一定ではないことが多い
    - 障害時に異経路になった場合などを考慮する必要あり

RPF Check

- ルーティングテーブルをlookupすることにより、パケットがspoofされていた場合には、それをはじくような仕組みを実装
- 本件に限らず、あらゆるspoofなパケットをはじくことが出来るので、非常に有効
- 顧客部分はけっこう痛い？
  - エッジは意外に非力なルータが多いのでは

Filtering

- 正規のBGPピア以外からのBGPパケットを受け付けなくしてしまう
  - 自分宛てのみに適応
    - Operationでがりがり頑張るしかない
    - CPUの上昇はそれほど無いはず
  - Transit Traffic 全てに適応
    - eBGP multihop などの場合には注意

Using Private Address

- アドレスをPrivate化してしまう
  - 外部から内部に攻撃されることはない
  - eBGPセッションはさすがに・・・
  - IFまで実施すると、tracerouteで丸見え
  - Tracerouteの戻りのパケットがprivate addressだと、途中でicmpがブロックされる可能性がある
    - RPF Checkにきっともひっかかる

BGP No Announcement

- BGPで使用しているアドレス等をそもそも広報しなければ、パケットはやってこないだろう
  - いまさら難しいよ・・・という人も多いのでは
    - そんな綺麗に昔からaddressingが出来ていないのが現実か (悲)
  - 広報されない＝外部から確認が出来ない
  - IXなどでは有効か

DNS (おまけ)

- iBGPもDNSから推測されることもある
  - 例） E1-2-0-gw1.otemachi.md5.ad.jp
  - CXsXo interface queue block の時もそうだった
    - GE-1-2.cisco01.md5.ad.jp なんかが狙われていた