L-049

インターネットに接続された装置を IP アドレス毎に評価する手法に関する一提案

Proposal of method of evaluating computer linked to the Internet of each IP address

山口 榮作[†] 鈴木 常彦[‡] 長谷川 明生[§]
Eisaku Yamaguchi Tsunehiko Suzuki Akiumi Hasegawa

1. まえがき

インターネットは目まぐるしい変貌を続け，商用利用のみならず，インフラに接続されることも珍しくなくなっている．普及が促進するに従い，実社会同様の詐欺行為や，それを助長する迷惑な攻撃は後を絶たない．通信プロトコルの問題や，基本サービスにおける脆弱さが指摘されているが，大規模ネットワークであるが故に，旧方式からの変化は難しい．例えばDNS一つを取ってみてもそれは明らかである．英国の "The Measurement Factory" の調査では，2005年4月の時点で約1.5% ものサーバが BIND4 や Windows NT 4.0 の Microsoft DNS である調査結果が報じられている [1]．ISC が最後に BIND4 を更新したのは2002年11月であるが，その時の README には次のように記されている．

> The official version of ISC BIND is now 9.1.0, or failing that, 8.2.3.
>
> This is ISC BIND 4.9.11, hoped to be the last of 4.*, which we are releasing since it has an important security bug fixed. **Other less important security bugs in BIND4 remain *unfixed*. You should not be running it. You have been warned.**

しかし，その2年半後であっても依然として BIND4 は生き残っているし，国内においても運用されている事例を確認している．

このように，ネットワークの健全化は非常にハードルが高いが，それ故に裾の底上げを行う努力をし続けると共に，攻撃者からの防御策の開発は必要不可欠である．

通信相手に応じて通信の仕方を制御できると，通信の許可・拒否だけでなく，攻撃者の疑いがある相手には通信コストの高いものとして動的に制限を加えたりすることもでき，結果として正当な通信相手への応答を向上させることができる．

本論文では，IP アドレス毎に通信相手を識別し，サービス品質管理をする情報を構築する手法を検討するとともに，基礎実験による評価結果を報告する．

2. クライアントの識別手法

2.1 Active OS fingerprinting

TCP/IP の規格は RFC で定義されているが，OS におけるプロトコルスタックの実装には差異があり，この差異は OS fingerprint と呼ばれている．あるホストから送信される TCP パケットを解析すると，その OS 固有の実装に依存した情報から，OS を推定することが可能である．

調査目的のためのパケットを送信し，相手からの応答を誘い，これを観察することで OS fingerprint を得る方法がある．これを Active OS fingerprinting[2] と呼ぶ．

Active OS fingerprinting を行う実装例としては，QueSO(Que Sistema Operativo)[3] や nmap[4] などが挙げられる．

2.2 Passive OS fingerprinting

Active OS fingerprinting と対照的に，相手からのパケットを待ち，到着したパケットをキャプチャすることで，OS fingerprint を得る方法がある．これを Passive OS fingerprinting[5] と呼ぶ．

Passive OS fingerprinting を行う実装例としては，p0f[6], pf[7], Siphon[8] などが挙げられる．

2.3 Application Banner

プリケーションの応答メッセージから OS やアプリケーションを検知するものである．具体的には，TELNET や SSH の応答，HTTP の HEAD リクエスト，FTP の SYST リクエストなどで OS やアプリケーションを特定することができる．図1は，NetBSD において SSH サービスを行う Application Banner の抜粋であるが，下線部に注目すると OS が NetBSD であることと，SSH の実装を確認することができる．

```
foo$ ssh -v 10.10.10.10
OpenSSH_4.4p1, OpenSSL 0.9.8e 23 Feb 2007
                    ...
debug1: Connecting to bar.example.jp [10.10
.10.10] port 22.
                    ...
debug1: Remote protocol version 2.0, remote
software version OpenSSH_4.4 NetBSD_Secure_Sh
ell-220061114
debug1: match: OpenSSH_4.4 NetBSD_Secure_Sh
ell-20061114 pat OpenSSH*
debug1: Enabling compatibility mode for pro
tocol 2.0
debug1: Local version string SSH-2.0-OpenSS
H_4.4
                    ...
Password:
```

図 1: NetBSD における SSH の Application Banner

2.4 TCP 3-way handshake control

TCP はセッション確立のために，3-way handshake を要するが，この時のパケット応答を制御すると，通信相

[†]愛知県立大学 情報化学部 情報システム学科
[‡]中京大学 情報理工学部 情報システム工学科
[§]中京大学 生命システム工学部 身体システム工学科

手の OS やアプリケーションによって応答が異なることが確認されている [9].

受動的な応答制御であるが，3-way handshake に対して割り込む形で能動的な制御の面もあり，複合的な位置づけである.

2.5 DNS check

DNS では IP アドレスやドメインの管理権限を持っている所からの，委譲関係が必須となっており，権限委譲を受けないと RR(Resource Record) の操作を有効に周知できない.

攻撃者は，より多くの踏み台を確保するため botnet[10] 等を多用する傾向があるが，その多くは動的割り当てを受けた IP アドレスであり，IP アドレスらしき数値を含む機械的に割り当てられたものが多く見られる傾向がある．また，Sender Policy Framework や Domain Keys などの新しい制御手法に関する RR など確認による評価結果も有効に活用できる情報が増えてきている.

3. 考察

Active OS fingerprinting では，相手がスキャンされていることを知る手がかりを残すことになるため，継続的な調査が結果に影響を及ぼす可能性が考えられる．OS fingerprint を偽装するためには，応答特性を変えるために，カーネルスペースにまで入り込んだ工作が必要であり，比較的偽装難易度が高いと言える.

Passive OS fingerprinting では，アクセスしようとしてくるホストに限定して，相手にスキャンされていると知られることなく，OS 情報を収集することができる．しかし有用性について検討した結果，確実に OS fingerprint が変化してしまう場合があることや，ユーザスペースからの偽装による霍乱が容易であることが確認できており，偽装の難易度が高いとは言いがたい [11].

Application Banner は，アプリケーションやライブラリに埋め込まれているメッセージであったり，設定ファイルに記述されているメッセージであるため，偽装が容易なことは自明である.

TCP 3-way handshake control は，前述の他の識別手法と比較して，識別判断基準が明確にはなっていないが，RFC に基づいた実装をしているか，手を抜いた実装をしているかの判別には非常に有効な新しい手法である．実験では SYN に対する沈黙において，OS やアプリケーションに応じて異なる timeout が生ずることを確認している．また，DNS や SMTP のように冗長性を持たせ fallback する事を前提としている場合には，(1)SYN に対する沈黙，(2)3-way handshake 後の FIN，(3)SYN に対する RST，の 3 通りの組み合わせによる fallback 特性から違いを見出すことができている．これは，SMTP の MX 配送のように優先度を持たせた fallback 制御の下での観察が有効と言える.

DNS check は，複雑な RR の調査は自ら DDoS を招く可能性があるため慎重に検討しなければならないが，相手の存在の評価の一判断項目として有効と考える.

これらの情報の評価結果を加工処理した上でデータベースに登録することにより，通信相手に応じた判断基準を提供することは，実処理における判断を軽減させることに有益と考えられる．しかし，SMTP における DNS ブラックリストがそうであるように，誤った情報が恒久的に登録されるとサービスの障害となってしまうこともある．情報の寿命管理と，有効な情報が途絶え難いような再調査のスケジューリング手法の検討が肝要と考えられる.

4. おわりに

IP アドレス毎に通信相手を識別するための材料について検討した．パケットの送受信を行う上で制限を課す必要があるかもしれない相手であるか否かの判断のためには，個々の手法の評価結果に応じた重み加重の検討や，通信サービスに応じた判断基準の使い分けなど，より詳細な調査検討が必要と言える．また，害のあるアプリケーションの特徴調査も必要である.

今後は，SMTP など通信サービスを限定した上で，接続してきた相手の有害さ・無害さの評価基準について検討したいと考えている.

参考文献

[1] The Measurement Factory : http://dns.measurement-factory.com/surveys/200504.html(2005).

[2] Ofir Arkin : Identifying ICMP Hackery Tools Used In The Wild Today, December 4, 2000, http://www.sys-security.com/archive/securityfocus/icmptools.html (2000)

[3] Jordi Murgo : Els Apostols, http://web.archive.org/web/19991004032416/http://apostols.org/projectz/queso/ (1998).

[4] Fyodor : Free Security Scanner For Network Exploration & Security Audits, http://www.insecure.org/nmap/

[5] Honeynet Project : Passive Fingerprinting, http://project.honeynet.org/papers/finger/

[6] Michal Zalewski : the new p0f, http://lcamtuf.coredump.cx/p0f.shtml

[7] Daniel Hartmeier, OpenBSD team : The Open BSD Packet Filter, http://www.openbsd.org/faq/pf/

[8] Subterrain Security Group : The Passive Network Mapping Tool, http://siphon.datanerds.net/

[9] 山口榮作, 鈴木常彦 : TCP Handshake 制御を利用した spam 対策システム, 大学情報システム環境研究, **Vol.8**, pp. 60–68 (2005).

[10] Paul Bächer, Thorsten Holz, Markus Kötter, Georg Wicherski : Tracking Botnets, http://www.honeynet.org/papers/bots/

[11] 山口，鈴木，長谷川 : 受動的な OS 特定法にみる，通信サービス品質改善の可能性に関する一考察, FIT2007(投稿中) (2007)