2015 年度 WIDE 秋合宿に関する報告

廣井 慧 (k.hiroi@ucl.nuee.nagoya-u.ac.jp),
高野 祐輝 (ytakano@nict.go.jp), 明石 邦夫 (k_akashi@jaist.ac.jp),
Camp-1509 プログラム委員会 (camp-1509-pc@wide.ad.jp),
加藤 朗 (kato@wide.ad.jp),
北口 善明 (kitaguchi@imc.kanazawa-u.ac.jp),
石原 知洋 (sho@c.u-tokyo.ac.jp), 高嶋 健人 (t.taketo1113@gmail.com)

1 序論

WIDE 合宿では毎回、ネットワークをプログラム委員（NetPC）が 1 から構築し、参加者に提供を行っている。このネットワークは、研究的なネットワークソフトウェアの運用であったり、実践的なネットワーク構築に関する教育を行う場であったりと、安定的に運用するというよりも、むしろ、実験的な場として運用される。

本稿では、2015 年 9 月 1 日 - 4 日にて行われた WIDE 秋合宿における合宿ネットワークおよび実験に関する報告を行う。本合宿では、ネットワーク構築に関しての実践的な教育を主眼とし、さらに、最先端技術の実験・実証についても行うという方針でネットワーク構築と運用を行った。

1.1 合宿ネットワーク構築と教育

ネットワーク構築や運用は、社会インフラを支える技術であるため、実際に動いているネットワーク機器を用いて実習を行うことは難しい。そこで、本合宿では、ネットワーク技術の初学者が実際に機器を設定し、ネットワークの構築と実運用を行えるような環境を用意した。2、3 節にて、合宿ネットワークを用いた教育及び、合宿ネットワークの構成について述べる。

1.2 合宿 PC によるネットワーク実験

本合宿では、合宿のプログラム委員が 3 つの実験を行った。

1 つめは、4 節で述べる、最新のソフトウェアを用いたネットワーク観測と可視化である。クラウド技術の発達とともに、ネットワーク観測及び可視化技術についても発展してきている。そのため、本合宿では、極力最新のソフトウェアを利用したネットワーク観測と可視化を行った。

2 つめは、Captive Portal を用いた合宿アンケート収集実験である。本実験は、合宿アンケートアナウンス及び収集の完全自動化を目指した。本実験の内容と結果については 5 節で述べる。

3 つめは、Happy Eyeballs 実証実験である。Happy Eyeballs は IPv6/IPv4 デュアルスタック環境下で、クライアントホストが接続遅延なくインターネッへ接続するための技術である。本合宿では、IPv6/IPv4 デュアルスタック環境を合宿参加者へ提供し、Happy Eyeballs の検証を行った。詳細な実験内容と結果については 6 節で述べる。

1.3 合宿参加者による実験

WIDE 合宿では、合宿参加者による実験も行い、NetPC が実験サポートを行っている。本合宿では 2 つの合宿参加者実験が行われた。

1 つめは、著者の加藤による、合宿でのトラフィックの記録である。本実験では、合宿参加者のネットワークトラフィックすべての保存を行っている。詳細な内容については 7 節にて説明する。

2 つめは、著者の北口、石原、高嶋らによる、デュアルスタックでのネットワーク接続性診断実験である。本実験では、北口が制作を行っているネットワーク接続診断ソフトウェアの実証実験を行った。

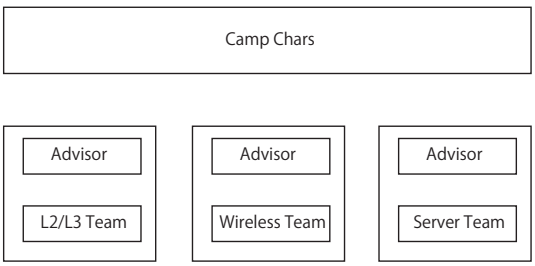

図 1　Camp-net チーム編成

本実験の詳細は、8 節にて述べる。

2　合宿ネットワーク

2.1　合宿ネットワーク方針

本合宿では、「研究論文もしくは標準化を目指すためのネットワーク実験」をテーマに幅広く実験を募集するとともに、モダンなネットワークツールを用いたネットワーク情報の可視化に取り組んだ。また、合宿ネットワークを構築する NetPC は、ネットワーク構築になれた人材が多く新人の参加が少ない傾向にあった。100 人以上が参加する WIDE 合宿は、イベントネットワークの構築経験として非常に有意義である。そこで本合宿では、NetPC 経験のない学生を多く募集し、新しい学生の育成に取り組んだ。

Net メンバは、図 1 に示す構成を取った。Advisor には NetPC への参加経験があり、L2/L3 や無線、サーバ構築に慣れた人にお願いした。L2/L3、Wireless、Server の各チームには、新しく参加した NetPC メンバの希望を聞き担当を決めた。そして、基本的に合宿ネットワーク構築は各チームのメンバが行い、Advisor は手を動かさない方針とした。

ネットワーク構築に慣れていない学生を主体とし合宿ネットワークを構築した場合、技術の習熟に時間が必要である。そこで、合宿ネットワークで提供するサービスや実験の環境構築には StarBED のサーバを利用し、時間をかけて合宿ネットワークの構築を行った。

2.2　ネットワーク構成

本節では、合宿にて構成したネットワークの概要について報告を行う。

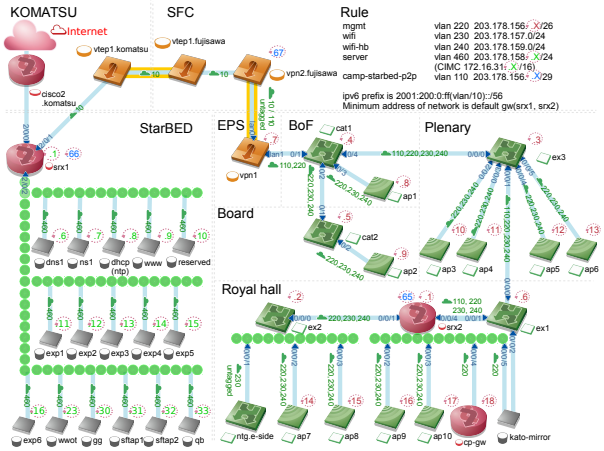

図 2　2015 年 WIDE 秋合宿ネットワーク図

2.2.1　対外接続

合宿地からの対外接続は、フレッツ網を用いた。インターネットへの到達性を確保するにあたり、2015 年 春合宿と同様 ISP 契約は行わず、NGN 網で配布される IPv6 アドレスを用いて WIDE BackBone (WIDE BB) と合宿地を接続した。フレッツ網から WIDE BB までは、NGN 網にて割当を受けた IPv6 アドレス 2408:212:6083:8200::/56 を用いて、L2TPv3 によるトンネリングを行った。L2TPv3 によるトンネリングは合宿地である松代ロイヤルホテルから藤沢 NOC 間で行った。

本合宿では、ほとんどのネットワークサービスおよび一部のトラフィックキャプチャ装置を StarBED に設置した。そのため、合宿地からの L2 ネットワークを StarBED まで延伸させる必要があった。そこで、L2TPv3 の終端点である藤沢 NOC から小松 NOC までを VXLAN によるトンネル接続を行い、合宿地から送信されるトラフィックは全て StarBED を経由する構成とした。

合宿地側のルータ (図 2 中の vpn1) で行った L2TPv3 の設定は、以下のとおりである。

```
interface l2tp0 description ”camp-1509”
interface l2tp0 tunnel 2408:212:6083:8201::1
2408:211:40e0:1701::1
interface l2tp0 l2tp manual local-id 1 remote-
id 2501
interface l2tp0 tcp-mss 1402
interface l2tp0 tcp-mss6 1382
```

藤沢 NOC から小松 NOC までに使用した VXLAN 設定は以下のとおりである。WIDE BB では、Multicast による通信が可能なため、VXLAN も Multicast を使用する設定とした。VXLAN Network Identifier (VNI) は 10 を使用している。

```
auto vxlan10
iface vxlan10 inet manual
vxlan-vni 10
vxlan-group 239.0.0.10
vxlan-dev eth0
post-up ifconfig $IFACE up
post-up ifconfig $IFACE mtu 1500

auto br10
iface br10 inet manual
bridge_ports vxlan10 eth1.10
bridge_stp off
bridge_fd 0
bridge_maxwait 0
```

2.2.2 Layer3 の構成

Layer 3 の構成は、合宿地内と StarBED でそれぞれ別のネットワークを構築した。

StarBED 内では、実験の一部と DHCP や DNS、WWW などのサービスを動作させた。StarBED 内のサーバは、NetPC だけではなく他の実験者が利用しているため、外部からの攻撃に対して考慮する必要があった。そのため、インターネット、合宿地の双方からのアクセスに対して Firewall を設置した。

もう一つの理由として、合宿地では Captive Portal 実験にてアンケートシステムを合宿期間中に動作させた。Captive Portal は HTTP でのアクセスをフックする構成上、ルータとして Captive Portal Gateway を設置する必要があり、これを StarBED に設置した場合、障害発生時の対応が難しくなる。障害発見、対応の容易さから本システムは合宿地で動作させるほうが望ましい。以上の理由により、StarBED と合宿地を別ネットワークで構成し、合宿地と StarBED の間は、Point-to-Point での接続とした。

表 1　使用した L2 スイッチ

ベンダー名	型番 x 個数
Juniper Networks	EX2200-48T-4G x 3
Cisco Systems	WS-C3560G-24PS-S x 2

表 2　VLAN 番号

VLAN ID	用途
110	StarBED、合宿地接続用
220	管理用
230	無線用
240	無線用 (IPv6 閉域網)
460	サーバセグメント

2.2.3 Layer2 の構成

本合宿で使用した L2 スイッチは表 1、VLAN とその用途は表 2 の通りである。ホットステージの段階では、合宿地内の L2 スイッチを EX2200 のみで構成していたが、アクセスポイントの電源確保が難しかったため、PoE が使用可能な Catalyst 3560 を 2 台投入した。また、合宿初日に kato-mirror へミラートラフィックを送るため、EX2200 (ex1) を SRX220H (srx2) の上に設置した。

2.2.4 無線ネットワーク

無線アクセスポイントは、11n に対応したものと 11a/g にのみ対応したもの含め全部で 11 台利用した。使用した無線アクセスポイントの種類は表 3 の通りである。接続するアクセスポイントによって利用可能な帯域が異なるため、アクセスポイントの配置位置はなるべく多くの人が 11n が利用可能と

表 3　使用した無線アクセスポイント

ベンダー名	型番 x 個数
Cisco Systems	AIR-AP1131AG-P-K9 x 5
Cisco Systems	AIR-CAP3502I-Q-K9 x 2
Cisco Systems	AIR-CAP3602I-Q-K9 x 1
Cisco Systems	AIR-RM1252G-P-K9 x 3

なるよう参加者が集まる位置に設置した。

これまでの合宿では、Wireless LAN Controller (WLC) を用いてアクセスポイントの制御を行っていたが、本合宿では実験として Captive Portal を使用するため WLC が利用できなかった。そのため、本合宿ネットワークでは WLC 使用時に用いる Lightweight アクセス ポイントではなく、従来の Autonomous アクセスポイントを使用した。アクセスポイントのチャネル割当も、各アクセスポイントに対して手動で設定を行っている。設定した各アクセスポイント毎のチャネル割当は表 4 の通りである。

合宿参加者へ提供する SSID は、wide2015 と wide2015-nohb を使用した。wide2015 は、今回行った実験の一つである Happy Eyeballs 用である。Happy Eyeballs で提供する IPv6 のインターネット到達性のないネットワークでは、合宿地での作業に不備がでる可能性があることも考慮し、IPv6 での到達性も持たせた wide2015-nohb も別途提供した。また、WIDE Web of Things Services にて大量の Bluetooth Low Energy デバイスを用いていたため、2.4Ghz 帯への影響を調査する必要があると考えた。そこで合宿 2 日目から、wide2015 を 5GHz 帯のみに変更し、2.4Ghz 帯は wide2015-2.4 の SSID を別途提供した。

2.2.5　サーバ

本合宿ネットワークは、提供するサービスをすべて StarBED 上に構築した。サービスの構築には、StarBED の Group-I (Cisco UCS200) を使用し、以下のサービスを提供した。

- DNS コンテンツサーバ

表 4　チャネル割当

部屋	AP 名	2.4GHz	5GHz
BoF	ap1	1ch	36ch
Board	ap2	6ch	44ch
Plenary	ap3	1ch	100+104ch
	ap4	11ch	108+112ch
	ap5	11ch	40ch
	ap6	6ch	52+56ch
Royal Hall	ap7	6ch	100+104ch
	ap8	11ch	108+112ch
	ap9	1ch	48ch
	ap10	6ch	52+56ch

- DNS キャッシュサーバ
- DHCP サーバ、NTP サーバ
- WWW サーバ

2.3　合宿を終えて

本合宿ネットワークは、これまで NetPC で活動してきた人に Advisor についてもらい、PC 経験のない学生主体で構築してもらい、WIDE メンバーの技術力向上を目指した。結果、これまでルータやスイッチ、サーバ構築の経験がない学生が 100 人規模のイベントネットワークを構築できたことは大きな成果である。

- 学生 A :ホットステージでも合宿当日でも Advisor がずっと近くにいてくださったので質問しやすくとても作業しやすかったです。また、同じグループの学生とも繋がりができたので今後にも活かせるのではないかと思います。
- 学生 B: 無線ネットワークの設定や運用について Advisor に指導していただきながら、環境構築を行えたのでとても良い勉強になりました。また HotStage や合宿中、Captive Portal の方に時間を割くことが出来たのも、Advisor にテキパキと作業の段取りをしていただいたおかげでした。

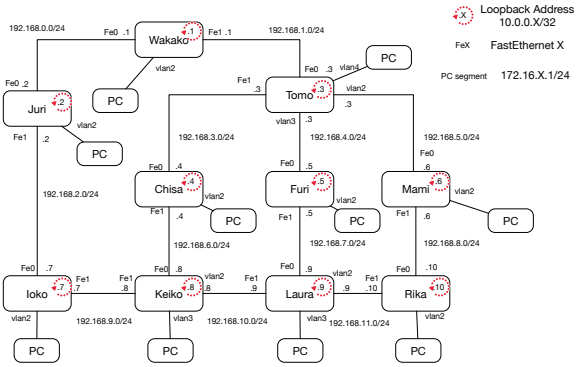

図 3　ハンズオンのネットワーク図

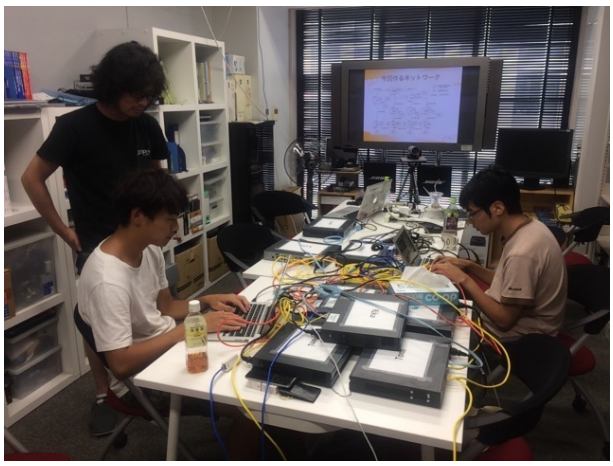

図 4　ハンズオンの様子

3　Net 教育

3.1　目的

本合宿では、WIDE 合宿で培った知見や、技術を若い学生に教えることで WIDE メンバー全体の技術を向上させることを目的の一つとした。しかし、経験のない学生が 1 週間という短いホットステージ期間で、様々な技術を習得するのは難しい。そこで、ホットステージ前にネットワーク構築に関する基礎的な技術を習得してもらうべく、北陸先端科学技術大学院大学と名古屋大学にてハンズオン形式の教育を行った。

3.2　ハンズオンの内容

使用機材は Cisco の 1812J であり、ハンズオンでの内容は以下のとおりである。

- シリアルケーブルでの接続方法
- ルータの初期設定
- Access Control List の設定
- OSPF の設定
- GRE によるトンネル接続

図 3 はハンズオンで、構築したネットワークである。ネットワークトポロジは、OSPF を動作させ障害発生時に自動的に経路が切り替わる様子を確認できるような構成である。

図 4 はハンズオン中の様子である。学生には複数台のルータを設定してもらい、各設定の意味やそれによる挙動などの確認をしてもらった。

3.3　学生の感想

- 学生 A: 私自身、ルータの設定をする経験が以前まで少なかったのでとても良い機会でした。初めてルータの設定触る学生も参加していたが最後まで取り組んでいて有意義な時間だったと思います。

4　最新のソフトウェアを用いたネットワーク観測と可視化

4.1　ログ監視・可視化について

本合宿でモダンなネットワークツールを用いたネットワーク情報の可視化に取り組んだ。過去の合宿では、MRTG や RDDtools など古くから利用され安定的に動作するツールが可視化ツールとして使われていた。しかしながら、クラウド技術や Web 技術の発達とともに、Web ブラウザ上で JavaScript のライブラリを用いたインタラクティブにログを表示するツールが発展してきた。そこで本合宿の実験として、ネットワーク機器のログ、サーバのログを可視化するために以下のツール群を用いた。

- Naigos Log Server (syslog 可視化)
- gri (スイッチのトラヒック情報可視化)
- Nagios Core (サーバ・サービス監視の可視化)

各ツールについてそれぞれ解説を行う。

4.2 Nagios Log Server

Nagios Log Server はオープンソースソフトウェア (OSS) である 3 つのソフトウェアを組み合わせた Nagios 社による商用パッケージソフトウェアである。可視化部分に Kibana, データストアに ElasticSearch, ログを受けるソフトウェアとして Logstash が利用されており、個々のソフトウェアを個別に導入しなくても一括でインストール可能である。

4.3 gri

gri (https://github.com/iij/gri) は株式会社インターネットイニシアティブによって作成され、OSS として公開されたネットワーク機器のトラヒック可視化ツールである。実装言語は Ruby で、ターゲットの IP アドレスや snmp コミュニティ名を設定に記述するだけで cron プログラムとして自動的にターゲット機器のデータの取得とグラフの表示を行う。

4.4 Nagios Core

Nagios Core は Nagios Log Server 同様に Nagios 社によって作成された監視ソフトウェアである。OSS の監視ツールとして世界的に有名なソフトウェアであり、サーバ自身の監視、並びにサーバ上で動作するサービス監視を行うソフトウェアとして利用される。

4.5 ログの可視化

Nagios Log server で可視化した例をいくつか解説する。図 5 は、camp-1509 で構築された WWW サーバ (https://www.camp.wide.ad.jp/camp/15autumn/) のアクセスログ解析になる。

アクセス元ホスト、HTTPD のステータスコード、GeoIP から求められる地図情報、アクセス元 OS やブラウザの種類など、HTTPD のログから取得可能な情報をわかりやすい形で可視化した例である。

次に、無線アクセスポイントからの情報を可視化した例を示す。本合宿ネットワークでは、無線 LAN のアクセスポイントとして、2.4GHz 帯と 5GHz 帯の 2 つが用意された。どちらの周波数ネットワークにどれだけのクライアントが接続していたかを時間帯に分けグラフにプロットしたものが図 6 となる。アクセスポイントごとに色を区分けすることにより、どの時間帯にどのアクセスポイントへどれくらいの人数が接続して以下も読み取ることが可能となる。

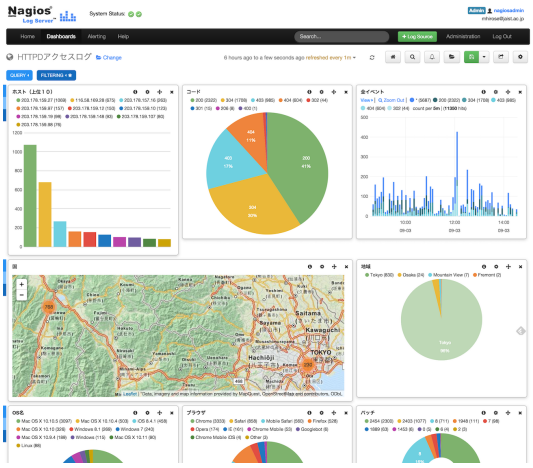

図 5 Nagios Log Server: camp web ログ解析

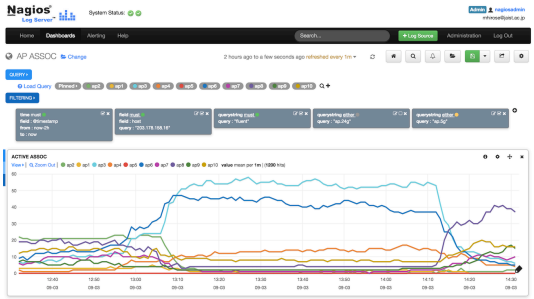

図 6 Nagios Log Server: 無線アクセスポイントログ解析

図 7 は、トラヒックデータとして gri を用いて可視化を行ったものである。gri では RRDtools を用いた可視化となり、シンプルにネットワークの bps, pps の実値、平均値、最大値を情報として表示することができる。

4.6 ログの送受信

ログの可視化を行うにあたり、ログの受信機構として fluentd を用いた。fluentd (http://www.fluentd.org/) は汎用ログ処理機構として実装されたミドルウェアであり、input プラグ

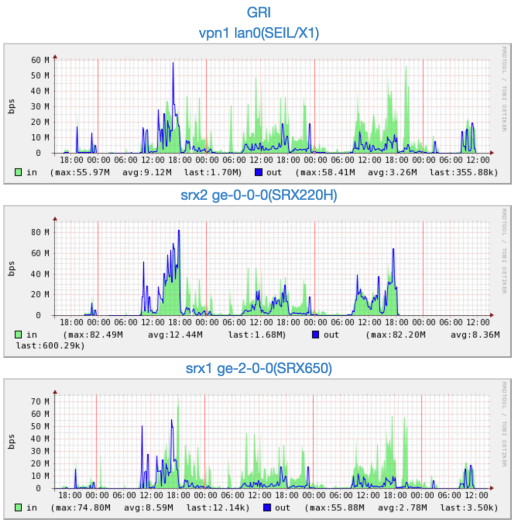

図 7　Gri: ネットワーク機器トラヒックデータ

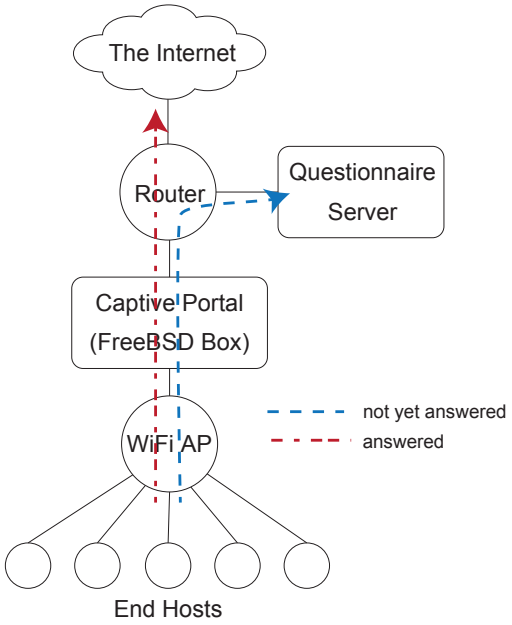

図 8　Captive Portal を用いたアンケート収集システム、ネットワーク構成図

インと output プラグインを組み合わせることにより、入力データと出力先を多く選択することができる。本合宿では、入力データとして、syslog, httpd のアクセスログ、nginx のアクセスログ、NetFlow、sFlow、無線 LAN の AP 情報などを取得し、出力先として ElasticSearch を用いデータのストアと表示に利用した。fluentd を用いることにより、データを正規化して扱うことができ、かつ fluentd を多段に動作させることにより処理に対してのスケールを担保することができる。

5　Captive Portal を用いた合宿アンケート収集システムの運用

本合宿では Captive Portal を用いて合宿アンケートの収集を行い、従来人手で行っていたアンケートの告知と収集を完全自動化した。

5.1　Captive Portal

Captive Portal とは、無線 LAN 接続時の認証機構であり、本機構を用いることで、ユーザの無線 LAN へのアクセスを細かくコントロールすることが出来る。一般的には、課金制の公衆無線 LAN などで用いられることが多く、クレジットカード決済を行ったユーザのみに接続を許可するなどの目的に利用される事が多い。

5.2　合宿アンケートと問題点

WIDE 合宿では、毎回、合宿終了時に参加者に対し、合宿内容についてのアンケートを行っており、このアンケートの収集は、参加者に、Web CGI ベースのフォームに記入してもらうことで行われている。Web のフォームに記入するのみで収集可能なこの方法は、非常に効率的であると思える。しかしながら、実際の運用では、アンケートの告知に大きな労力が支払われており、合宿中に頻繁に口頭でのアンケート告知を行うことで、アンケートの回収を実現している。この方法の問題点は、何度も告知を行う運営側の負担と、既にアンケートに答えた参加者の負担という 2 点に集約される。

5.3　Captive Portal を用いた合宿アンケート収集システム図

図 8 は、Captive Portal を用いたアンケート収集システムと、そのネットワーク構成図である。我々は、Captive Portal を実現するために、pfSense (https://www.pfsense.org/) というオープンソースのパケットフィルタリング、ファイアウォールソフトウェアを利用した。pfSense は、FreeBSD の

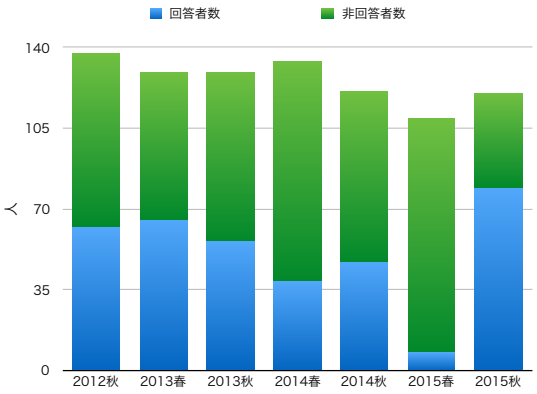

図 9　合宿参加者とアンケート回答数の遷移

パケットフィルタリングシステムである pf を用いているため、FreeBSD 上でのみ動作し、図 8 中の Captive Portal (FreeBSD Box) が、pfSense を実行している機器である。この、Captive Portal では、End Hosts から WiFi AP 経由で届いたパケットを、合宿アンケートサーバか、インターネットのどちらかへ転送する。すなわち、合宿アンケートに答えていなかったら、合宿アンケートサーバへ、既に答えていたらインターネットへと転送を行う。

Captive Portal を実現するためには、合宿アンケートに答えたかどうかという判断を行う必要があるが、これは、MAC アドレスによる認証を用いて実現した。すなわち、合宿アンケートページに回答した端末からの MAC アドレスを記憶しておき、記録された MAC アドレスからの通信の場合は、インターネットへと転送を行う。したがって、本システムでは、複数の端末を用いてアクセスする場合、合宿アンケートに複数回回答しなければならないという制限がある。ただし、合宿アンケートに一度答えると、回答結果がアンケートサーバに保存されているため、二端末目以降は、合宿アンケートページで HTML フォームのボタンをクリックするのみで良い。

5.4　効果と結論

本合宿では、3 日目の 18 時、夕食開始時に Captive Portal システムを稼働させた。これまでの合宿ならば、アンケートのアナウンスを口頭で何度も行っていたが、本合宿は一度も口頭でのアナウンスを行わなかった。それにもかかわらず、本合宿でのアンケートは合宿参加者数 120 名中、79 名 (65.8 %) からもの回答があった。図 9 は、2012 年以降の合宿参加者とアンケート回答者数の遷移を示している。この図からわかるように、本合宿でのアンケート回答数は、ここ数年で最も多いことがわかる。この結果より、本システムは合宿アンケート告知と回収の自動化のみならず、回収率に関しても効果が有ることが立証された。

6　Happy Eyeballs 実証実験

6.1　Happy Eyeballs とは

Happy Eyeballs [3] は、IPv4/IPv6 デュアルスタックネットワーク環境下で、エンドホストが適切にインターネット通信を行うための技術である。

IPv6 が定義された当初は、getaddrinfo() 関数を利用して名前解決を行った後、getaddrinfo から得られるリンクリストに対して、connect() 関数で接続する方法が推奨された [4]。しかしながら、この方法では特定の IPv6/IPv4 のデュアルスタック環境下で接続時の遅延が著しくが大きくなることから、RFC 6555 にて Happy Eyeballs が定義された。

Happy Eyeballs では、同時に IPv6、IPv4 へと接続を行い、早く接続された方のコネクションを優先的に採用する。このようにすることで、IPv6/IPv4 デュアルスタック環境下で、IPv6 アドレスを用いてインターネットへ接続出来ない場合でも、接続遅延なく通信を行うことが出来る。

6.2　本合宿での取り組み

本合宿では、NTT の提供するフレッツネットワークを模した環境を、合宿参加者に提供した。フレッツネットワークは、IPv6/IPv4 のデュアルスタックネットワークであり、利用者に対して IPv6 と IPv4 のグローバルアドレスを提供する。しかしながら、フレッツネットワークは IPv6 ネットワークを閉域網として利用するため、IPv6 アドレスを用いてインターネット接続を行うことは出来ない。

本合宿でも、フレッツネットワークと同様に、合宿参加者に対して IPv6 と IPv4 のグローバルアドレスの配布を行った。ただし、IPv6 パケットは上

位のルータでドロップするように設定し、フレッツネットワークと似た、IPv6 での接続性がない状況を再現した。

6.3 結果と結論

本実験を行った結果、合宿参加者から 2 つ報告が得られたのでここに報告する。

1 つめの報告は、ssh での接続がスムーズに行えなく、大変苦労したというものである。これは、ssh クライアントは、6.1 節で説明した、シーケンシャルに接続を行う方法で実装されているからと考えられる。そのため、IPv6 アドレスが割り当てられている ssh サーバに対して接続を行った場合、著しい接続遅延が発生したと予測される。

2 つめは、幾つかのブラウザで、特定サイトにアクセスした際の見え方が変わっているとの報告である。(Chrome ではアクセス出来ないが、Firefox、Safari ではアクセスできるなど) これについては現象を確認したのみで、原因までは特定できていない。

これら以外にも、いくつかのサイトにアクセス出来ないなどの報告が寄せられた。

本実験を行う前に、合宿参加者全員に対して、IPv6 での接続が出来ないので留意する必要があると、再三アナウンスを行った。また、WIDE 合宿はインターネット技術のスペシャリストが集う研究会議である。それにもかかわらず、IPv6/IPv4 デュアルスタック環境下で IPv6 でのインターネット到達性を不可にした場合、その原因特定までに、少なくない時間がかかってしまっている。これはすなわち、エンドユーザにとって、現状のデュアルスタック環境か、Happy Eyeballs を正しく実装していないアプリケーションのどちらか、もしくは両方に大きな問題を抱えているという証左である。したがって、今後は、ネットワークインフラとエンドアプリケーション、両方含めての対応が必須となると考えられる。

7 合宿でのトラフィックの記録

WIDE Project の合宿は、参加者が一般のインターネットユーザとは多少性質が異なる可能性が高いとは言え、トラフィックサンプルを取得し、記録しておくことは、現在の様々なトラフィック計測に役立つだけではなく、長期的なインターネットトラフィックの傾向の変化に関しても確認することができることが可能な数少ない機会である。そこで、2014 年秋合宿より、合宿ネットワークの根本の Layer-2 Switch の port mirroring を用いてトラフィックを記録することにした。

記録に用いているのは、Shuttle 社の DS-81 という小型の Bare Bone PC である。本体は比較的頑丈に設計されているため、重量はややあるものの、TDP 65W までの Intel CPU をサポートしており、Realtek 8111C による Gigabit Ethernet を 2 ポート有している。そこで、この筐体に Intel i5-4590S (4 core, 3GHz, turbo boost 3.7GHz) に 8GB メモリを搭載し、さらに 2.5" 1TB 7200RPM 9.5mm 厚の HGST 製の disk を装備し、NetBSD6.1 を install したものを準備した。さらに、

- Gigabit Ethernet ポート 1 は、DHCP でアドレスを取得し、管理用のアクセスを提供する
- Gigabit Ethernet ポート 2 は、取得すべきトラフィックを受信するポート

のように接続されることを仮定し、必要な設定を行った。

計算機が boot すると、毎時 0 分、15 分、30 分、45 分のいずれかになるまで待機し、それ以降は 15 分毎に別なファイルになるようにしながら traffic を取得し、gzip された pcap 形式でファイルに記録するように tcpdump を起動するような script を準備し、/etc/rc.local に記述した。このため、電源を入れ、Layer-2 switch の mirror port を Gigabit Ethernet ポート 2 に接続するだけで、トラフィック情報の記録が可能である。

過去 3 回の合宿で取得されたデータの概要は以下に示す。容量は、gzip された状態での pcap ファイルの合計値である。基本的に全てのパケットを記録しているため、IP 以外の LLDP や STP などような L2 プロトコルのパケットも含まれている。

このデータはプライバシ情報や機密情報を含んで

表 5　合宿での traffic capture

合宿	容量	ファイル数	総パケット数
2014 秋	203GB	290	331 Million
2015 春	320GB	262	771 Million
2015 秋	310GB	288	1,400 Million

いる可能性があるため、アクセスには、解析計画書を提出頂き、審査している。また、計算機からデータを持ち出すことは禁止しており、自分で作成した解析プログラムを保管用サーバ（DS-81 ではない）に移植して、実行することになっている。

今現在はあまり多くのアクセス要求はないが、継続的に記録を行うことによって、インターネットでのトラフィックの状況の変化を観測することができるものと期待している。

8　デュアルスタックでのネットワーク接続性診断

8.1　実験の背景

キャンパスネットワークやイベントネットワークの運用において無線アクセスネットワークを提供する場合、ユーザから「つながらない」とのクレームを受ける時がある。「つながらない」という現象の原因としては、提供されるネットワークにおける物理的な障害の他に、エンドユーザの端末に付与されるアドレスの問題や DNS における名前解決の不具合など、様々な要因が考えられる。このような「つながらない」状況の問題点を突き止める場合には、ユーザ側からのネットワーク観測が有効であるが、ユーザからは得てして「つながらない」という漠然とした状況しか得られないものである。

一方、ネットワークの運用者側からの対応としては、ネットワークや提供サービスの状態を監視する手法は多く存在しており、Nagios[*1]などのオープンソースアプリケーションが一般的に利用されている。ただし、これらの手法では、サーバの挙動や IP

[*1] http://www.nagios.org/

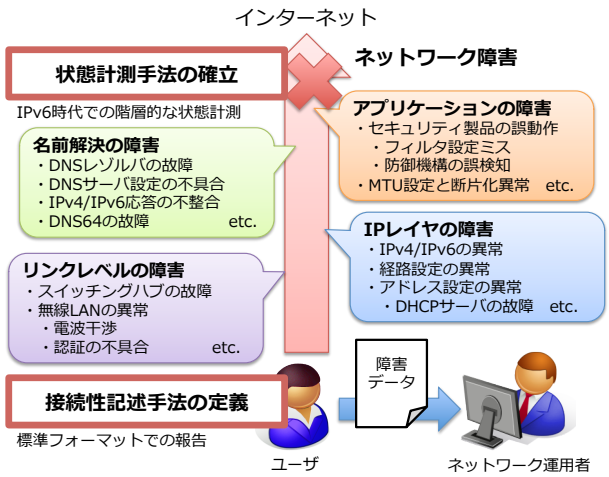

図 10　ユーザ視点によるネットワーク状態計測手法のイメージ

的な到達性の確認が主となるため、ユーザが被っている障害の状態やサービス障害点を確認することが困難であった。

さらに、今後利用が加速すると予想される IPv6 を利用したデュアルスタックネットワークでは、IPv4 と IPv6 それぞれの挙動を把握する必要があり、問題発生時の障害点の把握が一層難しくなることが予想されている。特に、クライアント OS 上においてドメイン名と IP アドレスの変換を行う DNS レゾルバの実装に関してはクライアント OS 毎に異なる場合があることが示されている [1]。

このように、ネットワークシステムの運用上における課題を解決するために、我々はエンドユーザが利用している実環境からの状態観測情報をネットワーク運用者に的確に伝える手法の確立が必要と考えている。図 10 に提案するネットワーク状態計測手法の概要を示す。今回の実験では、ネットワーク運用者が迅速に問題点を把握できる手法の確立を目的とした、ユーザ視点におけるネットワーク状態の観測手法に関して、プロトタイプ実装により評価を行う。

8.2　評価項目の定義

我々の提案手法では、ユーザ環境においてネットワーク障害点を検出する手段を、OSI 参照モデルの

ようにレイヤに分けて整理することで、迅速な障害把握を実現するシステム構築を目指している。

計測レイヤは、ネットワーク運用のモデル化に必要な概念として考えており、ネットワーク状態を階層的に表現し、それぞれのレイヤにて発生しうる障害を体系的に表現するために用いる。インターネットは OSI 階層モデルと同様に TCP/IP 階層モデル（4 層）により通信プロトコルが整理されているので、このモデルに沿う形で整理を進め、これまでの我々の取り組み [5] を元に 6 層の計測レイヤを定義している。

今回の実験では、具体的な評価項目を表 6 のように定義した。各層における障害検知の評価項目は boolean 型（bool）として定義しており、障害判断の大項目として利用することを想定している。information 型（info）として定義した評価項目は、発生している障害の詳細を解析するためや次のレイヤにおける評価項目を実施する判断を決定するためなどに用いる項目である。今回は、プロトタイプ実装可能な項目を中心に列挙している。

8.3 実験の内容

図 11 に我々が提案するネットワーク状態計測手法の概要を示す。今回はこの中で、Mac OSX 上で動作するシェルスクリプトとして評価項目を実行するプログラムを実装した。計測結果を JSON 形式でローカルに保存し、情報収集サーバに対して HTTP を利用して送信する仕組みとしている。計測は、定義した計測レイヤの下位層から順に実施し、上位層で実施する評価項目を選択している。具体的には、インタフェース層にて IPv6 アドレスが設定されていないことが確認できた場合、ローカルネットワーク層以降にて IPv6 通信を用いた評価項目を実施しない仕様としている。

また、下位層から実施する一連の計測に対してキャンペーン ID を設定し、計測元を端末のインタフェース（MAC アドレス）にて識別することとした。識別子として端末 ID ではなくインタフェース ID（MAC アドレス）を選択した理由は、複数インタフェースを有する端末が主流であることが挙げられる。IETF における mif WG においても、端末が

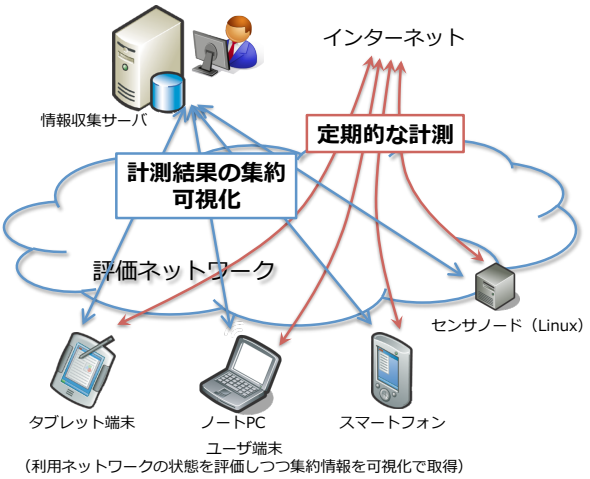

図 11　提案手法の計測イメージ

持つ複数のインタフェースを制御する仕組みに関して議論が行われており、その中にてインタフェース毎に DNS サーバを選択することが標準化されている [2]。従って、本提案手法においても、計測対象とするインタフェース毎に上位層の計測を選択することが肝要と言える。

情報収集サーバでは、収集した計測データをネットワーク運用者に的確に提示するための可視化の実装を行った。図 12 に可視化画面の一例を示す。プロトタイプ実装では、JSON 形式の計測結果を元に Web ブラウザにて成功を緑、失敗を赤で表現する可視化を行っている。継続的な計測を基にした時系列表示など、様々な表示方法を今後検討し、有効な表現方法を実現したいと考えている。

9 結論

本合宿では、ネットワーク技術の実践的教育に重点を置いた。そのため、スイッチなどの設定は学生などの初学者が主に担当し、ネットワーク構築の技術者や研究者は、Advisor という形で参加した。これは、運用、環境構築で失敗してもよいという実験的な場である WIDE 合宿であるからこそ達成できたことである。教育については、論文何本といったわかりやすい指標がある研究とは違い、成果が出るのは、教育を受けたものが十分に成長する 10 年、

図 12　可視化画面例

あるいは 20 年後であり、ここで定量的な評価を下すことは難しい。しかしながら、第一線で働く技術者・研究者に直接、マンツーマンに近い形での指導を受けられるのは、WIDE 合宿以外で出来るところは多くはないだろう。

また、本合宿では、教育以外にも最新技術の検証・評価を行うことを目的とし、最新技術を用いたログ可視化と監視システムの実験 Captive Portal を用いたアンケート回収システムの実験、Happy Eyeballs 実証実験という、3 つの実験を合宿プログラム委員が行った。これら実験についても、主に初学者が調べ実装を行った。

これら実験は、どちらかと言うと研究におけるフェーズで言うと、サーベイ段階のものである。実際の研究では、さらにここから、新規技術の提案と、既存技術との評価を行う必要があるが、初学者にとっては既存技術を習得するのも難しく感じることが事が多いだろう。実際に、本実験では、かなりの試行錯誤が行われて実施された。ネットワークシステムや実装に関する研究では、仮説を立て試行錯誤を行い、試行回数を重ねるのが何よりも重要である。本合宿で得た経験は、研究や技術開発を行う際に試行錯誤するために、必ず役に立つはずであり、今後に活きていくだろう。

一方、本稿著者の加藤、北口、石原、高嶋らによる実験は、研究のプロが行うものであり、論文化などを目指した研究のための実験である。特に、これら実験は本合宿以前から継続的に行われている実験であることから、研究論文とするためには、継続時な実験が必要であることがわかる。

本稿を読むとわかるように、教育や実験内容も WIP 的な内容であり、成功したと言うには定量的な評価が圧倒的に足りない。一般的に、本稿のような報告書では、成功したとして結論を結ぶことが多く、そうする方が容易であるが、本稿ではあえて、成功したと結論付けない。教育や、実験的研究が成功したかどうか分かるのは、10 年、20 年後のことであるため、その行く末を 10 年、20 年後に判断していい欲しい。

参考文献

[1] H. Hazeyama, R. Hiromi, T. Ishihara, and O. Nakamura. *Experiences from IPv6-Only Networks with Transition Technologies in the WIDE Camp Autumn 2012*, October 2012. draft-hazeyama-widecamp-ipv6-only-experience-02.txt.

[2] T. Savolainen, J. Kato, and T. Lemon. Improved Recursive DNS Server Selection for Multi-Interfaced Nodes. RFC 6731 (Proposed Standard), December 2012.

[3] D. Wing and A. Yourtchenko. Happy Eyeballs: Success with Dual-Stack Hosts. RFC 6555 (Proposed Standard), April 2012.

[4] 萩野純一郎. IPv6 ネットワークプログラミング. アスキー, 2003.

[5] 北口善明, 石原知洋, 高嶋健人, 田川真樹, 田中晋太朗. Raspberry pi を用いた無線ネットワーク状態評価手法の提案. 情報処理学会研究報告, 第 2014-IOT-25 巻, pp. 1–6. 情報処理学会 IOT 研究会, May 2014.

表 6　定義した計測レイヤと評価項目

階層名	型	種別	評価項目
datalink	bool	共通	インタフェース UP：OSI 参照モデルにおける Layer2 におけるリンクアップを確認
	info	共通	インタフェースタイプ：インタフェースのタイプを示す情報（Ethernet, Wi-Fi, LTE など）
	info	共通	インタフェース MTU：インタフェースの MTU 情報
	info	共通	MAC アドレス：インタフェースの MAC アドレス（識別子として利用）
	info	有線	メディアタイプ：オートネゴシエーション機能等で決定される通信速度や通信モードの情報
	info	無線	サービスセット識別子 (SSID)：利用している無線ネットワークの SSID 情報
	info	無線	チャンネル：利用している無線ネットワークのチャンネル情報
	info	無線	信号受信強度 (RSSI)：利用している無線ネットワークの RSSI 情報
	info	無線	ノイズ信号強度：受信しているノイズの信号強度情報
	info	無線	データレート：利用している無線ネットワークのデータレート情報
interface	info	IPv4	インタフェースの IPv4 設定：対象インタフェースの IPv4 における設定情報
	bool	IPv4	IPv4 自動アドレス設定 (DHCP)：DHCP によるアドレス設定を確認
	info	IPv4	IPv4 アドレス：インタフェースに設定された IPv4 アドレス情報
	info	IPv4	ネットマスク：インタフェースに設定された IPv4 ネットマスク情報
	info	IPv4	IPv4 デフォルトルータ：インタフェースに設定された IPv4 のデフォルトルータ情報
	info	IPv4	IPv4 ネームサーバ：対象インターフェースの自動アドレス設定にて取得した IPv4 ネームサーバ情報
	info	IPv6	インタフェースの IPv6 設定：対象インタフェースの IPv6 における設定情報
	info	IPv6	IPv6 リンクローカルアドレス：インタフェースに設定された IPv6 のリンクローカルアドレス情報
	bool	IPv6	IPv6 自動アドレス設定 (SLAAC/DHCPv6)：SLAAC および DHCPv6 によるアドレス設定を確認
	info	RA	RA のフラグ情報：インタフェースで受信する RA に設定される O フラグ、M フラグの情報
	info	RA	RA のプレフィックス情報：インタフェースで受信する RA に設定されるプレフィックス情報
	info	RA	RA のフラグ情報：インタフェースで受信する RA に設定されるプレフィックス情報のフラグ情報
	info	IPv6	IPv6 アドレス：インタフェースに設定された IPv6 アドレス情報
	info	IPv6	IPv6 デフォルトルータ：インタフェースに設定された IPv6 のデフォルトルータ情報
	info	IPv6	IPv6 ネームサーバ：対象インターフェースの自動アドレス設定にて取得した IPv6 ネームサーバ情報
localnet	bool	IPv4	IPv4 デフォルトルータ到達性：IPv4 のデフォルトルータへの到達性を確認（ping）
	info	IPv4	IPv4 デフォルトルータ往復遅延：IPv4 デフォルトルータへの往復遅延時間情報
	bool	IPv4	IPv4 ネームサーバ到達性：IPv4 ネームサーバへの到達性を確認（ping）
	info	IPv4	IPv4 ネームサーバ往復遅延：IPv4 ネームサーバへの往復遅延時間情報
	bool	IPv6	IPv6 デフォルトルータ到達性：IPv6 のデフォルトルータへの到達性を確認（ping）
	info	IPv6	IPv6 デフォルトルータ往復遅延：IPv6 デフォルトルータへの往復遅延時間情報
	bool	IPv6	IPv6 ネームサーバ到達性：IPv6 ネームサーバへの到達性を確認（ping）
	info	IPv6	IPv6 ネームサーバ往復遅延：IPv6 ネームサーバへの往復遅延時間情報
globalnet	bool	IPv4	外部サーバへの IPv4 到達性：外部サーバへの IPv4 による到達性を確認（ping）
	info	IPv4	外部サーバへの IPv4 往復遅延：外部サーバへの IPv4 の往復遅延時間情報
	info	IPv4	外部サーバへの IPv4 通信経路：外部サーバへの IPv4 の通信経路情報（traceroute）
	info	IPv4	外部サーバへの IPv4 パス MTU：外部サーバへの IPv4 通信路における最大 MTU サイズの値
	bool	IPv6	外部サーバへの IPv6 到達性：外部サーバへの IPv6 による到達性を確認（ping）
	info	IPv6	外部サーバへの IPv6 往復遅延：外部サーバへの IPv6 の往復遅延時間情報
	info	IPv6	外部サーバへの IPv6 通信経路：外部サーバへの IPv6 の通信経路情報（traceroute）
	info	IPv6	外部サーバへの IPv6 パス MTU：外部サーバへの IPv6 通信路における最大 MTU サイズの値
dns	bool	IPv4	IPv4 による A レコード名前解決：IPv4 を用いた FQDN に対する A レコードの名前解決を実施
	bool	IPv4	IPv4 による AAAA レコード名前解決：IPv4 を用いた FQDN に対する AAAA レコードの名前解決を実施
	bool	IPv6	IPv6 による A レコード名前解決：IPv6 を用いた FQDN に対する A レコードの名前解決を実施
	bool	IPv6	IPv6 による AAAA レコード名前解決：IPv6 を用いた FQDN に対する AAAA レコードの名前解決を実施
	bool	dual	OS リゾルバによる名前解決：利用する OS のリゾルバに対する名前解決（ANY）を実施
	info	dual	OS リゾルバによる名前解決オーダ：利用する OS のリゾルバに対する名前解決（ANY）の結果順序
web	bool	IPv4	外部ウェブサーバ (IPv4) への HTTP 通信：外部ウェブサーバへの IPv4 による HTTP 通信を確認
	info	IPv4	外部ウェブサーバ (IPv4) への HTTP 通信速度：外部ウェブサーバへの IPv4 による HTTP 通信速度情報
	bool	IPv6	外部ウェブサーバ (IPv6) への HTTP 通信：外部ウェブサーバへの IPv6 による HTTP 通信を確認
	info	IPv6	外部ウェブサーバ (IPv6) への HTTP 通信速度：外部ウェブサーバへの IPv6 による HTTP 通信速度情報
	info	dual	Happy Eyeball の挙動：利用する OS が提供する Happy Eyeball API の挙動を確認
	info	IPv4	透過型プロキシサーバの検出 (IPv4)：IPv4 通信路に存在する透過型プロキシサーバの検出
	info	IPv6	透過型プロキシサーバの検出 (IPv6)：IPv6 通信路に存在する透過型プロキシサーバの検出