

お客さま各位

KDDIインターネット イーサエコノミー v6プラス YAMAHA RTX810 固定IP8個 設定例

拝啓 ますますご清栄のこととお慶び申し上げます。
平素は格別のご厚情を賜り、厚く御礼を申し上げます。
本資料は、KDDIインターネット イーサエコノミー サービスをご利用いただく際のルータ設定(例)
を記載した資料となります。

敬具


-記-

1. 設定条件	2ページ
2. 確定通知書	3ページ
3. ネットワーク構成	4ページ
4. 設定について	5ページ

以上

【サービスに関するお問い合わせ】

法人お客さまセンター

 0077-7007 (無料) 0120-921-919 (無料)

※受付時間：9:00~18:00（土・日・祝日・年末年始を除く）
※上記フリーコールは、携帯電話・PHSからもご利用いただけます。

1. 設定条件

■ イーサエコノミーのご利用メニュー

- v6プラスの固定IP8個(/29)をお申し込みいただいた場合になります。

■ ルーターについて

- YAMAHA製 RTX810 の設定例となります。
- RTX810 の対応ファームウェアは、11.01.01 以上となります。
- v6プラスは、IPIP機能(IPv4 over IPv6)を使用します。
- RTX810 以外のルーターでも、IPv6でIPIP機能(IPv4 over IPv6)に対応している機種であれば、利用可能です。

※インターフェース等の情報は、機種毎に異なるので、変更頂く必要がございます

[参考] <http://www.rtpro.yamaha.co.jp/RT/ipv6/>

■ フレッツ光 回線について

- 本設定例は、光ネクスト ファミリー系(100M、200M、1Gbps) または マンション系(100M、200M、1Gbps) を対象としています。
- ひかり電話の利用は“なし”としています。

[参考]

- “光ネクスト”で“ひかり電話 利用なし”の場合、IPv6 アドレス(プレフィックス)の取得方法は、RA となります。
- “光ネクスト”で“ひかり電話 利用あり” 且つ ホームゲートウェイを利用しない場合、IPv6 アドレス(プレフィックス)の取得方法は、DHCPv6-PD となります。
- “光クロス”で レンタルルーター(10GBASE-Tを備えたホームゲートウェイ)を利用しない場合、IPv6 アドレス(プレフィックス)の取得方法は、DHCPv6-PD となります。

2. 確定通知書

ご利用開始日前までに、KDDI から設定に必要な情報を記載した情報(「確定通知書」といいます)を案内いたします。

■ 確定通知書サンプル:【v6プラス(固定IP8個)】(抜粋)

課金開始予定日	2021/4/1(木)	
お客様受付番号	N123456789 KDDIインターネットイ・エコミ・光ネット v6プラス/29	
お客様名 (エンドユーザー)	KDDI株式会社様	
利用場所都道府県/事業所名	東京都/v6プラス固定IP事業所	
割当IPアドレス (WAN側) 割当IPアドレス (LAN側)	IP Unnumbered 106.185.156.0/29	
お客様回線ID (フレッツお客さまID)	CAF1234567890	
ルータ設定情報	KDDI側ルータIP(トンネル接続)	2001:db8:225:100::65
	お客様ルータ(IPv6インターフェイスID)	006a:b99c:0000:0000
	設定ID	N123456789
	設定パスワード	CAF1234567890
	設定サーバ(URL)	http://fcs.enabler.ne.jp/update
DNS	プライマリホスト名	-----
	プライマリIPアドレス	-----
	セカンダリホスト名	-----
	セカンダリIPアドレス	-----
	参照用DNS プライマリ	※DHCPv6にて自動取得
	参照用DNS セカンダリ	※DHCPv6にて自動取得

※赤字の情報が、ルータの設定で必要となります。

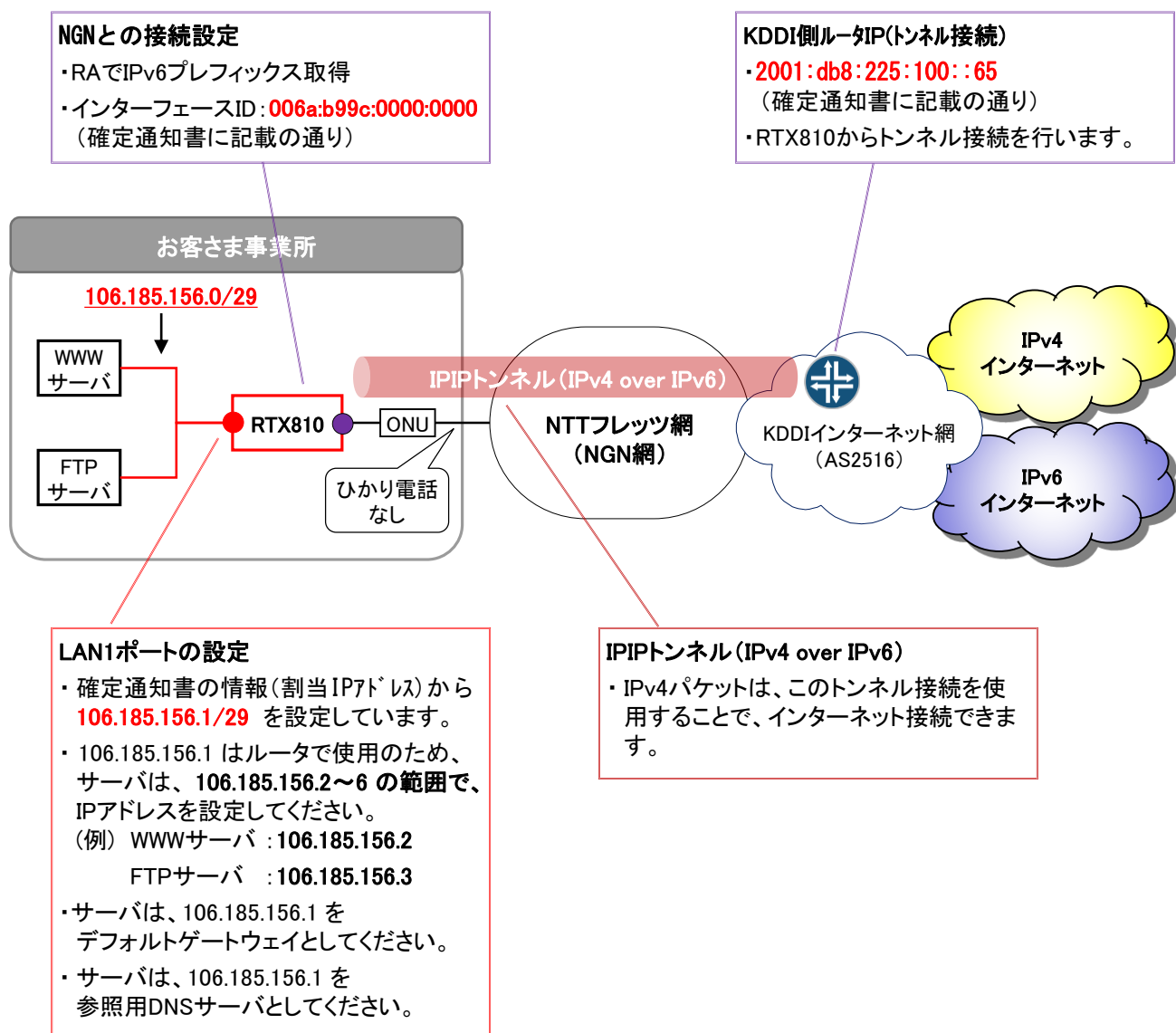
■ 項目の説明

割当IPアドレス(LAN側)	KDDIからお客さまに割り当てを行ったグローバルIPv4アドレスです。
お客様回線ID (フレッツお客さまID)	ご利用いただくフレッツ光回線のIDになります。 ※「v6プラス」は、お申し込みいただいたフレッツ光回線でのみご利用可能です。
KDDI側ルータIP(トンネル接続)	IP-IP方式によるトンネリング(IPv4 over IPv6)接続を行う宛先アドレスです。
お客様ルータ(IPv6インターフェイスID)	お客さまルータに設定いただくIPv6アドレスのインターフェイス識別子になります。 (IPv6プレフィックスは自動取得となります)
設定ID 設定パスワード 設定サーバ(URL)	NTTの局内工事等で、IPv6プレフィックスが変更となる場合が稀にございます。 この時、NTTシステムから弊社システムにIPv6プレフィックス情報が伝達されます。I 弊社システムで、Pv6プレフィックスと割当IPアドレスの紐づけ情報を更新することで、 IPv6プレフィックス変更後も、IPv4通信が利用できるようにいたします。 (情報更新されるまでの間(数分ほど)、IPv4通信が利用できなくなります) お客さまのご利用回線から設定サーバにアクセスし、ID・パスワードを入力することで、 即時に弊社システムの情報更新が行われるため、IPv4通信が利用できなくなる時間を 短くすることが出来ます。 ※ご利用のフレッツ光回線から IPv6 でアクセスいただく必要がございます。
参照用DNS プライマリ 参照用DNS セカンダリ	参照用DNSサーバの情報は、DHCPv6にて取得してください。

3. ネットワーク構成

■ 通信要件

- ・サーバには、グローバルIPv4アドレスを設定してください。
- ・サーバからインターネットへの接続は、可能とします。
- ・IPv4インターネットからサーバへの接続は、可能とします。(WWW、FTP)
- ・IPv6インターネットからサーバへの接続は、拒否します。



4. 設定について(1/2)

```

# 【経路設定】
ip route default gateway tunnel 1

# 【NGNとの接続設定】
ngn type lan2 ntt
ipv6 prefix 1 ra-prefix@lan2::/64
ipv6 lan1 address ra-prefix@lan2::006a:b99c:0000:0000/64
ipv6 lan1 rtadv send 1 o_flag=on
ipv6 lan1 dhcp service server
ipv6 lan2 dhcp service client ir=on

# 【LAN設定】
ip lan1 address 106.185.156.1/29

# 【IPIPトンネル設定】
tunnel select 1
tunnel encapsulation ipip
tunnel endpoint address 2404:9200:225:100::65
ip tunnel secure filter in 200030 200031 200032 200039 dynamic 201 202
ip tunnel secure filter out 200099 dynamic 200080 200081 200082 200083 200084 200085 200098 200099
ip tunnel nat descriptor 1
ip tunnel tcp mss limit auto
tunnel enable 1

# 【IPv4フィルタ設定】
ip filter 200030 pass * 106.185.156.0/29 icmp * *
ip filter 200031 pass * 106.185.156.2 tcpflag=0x0002/0x0fff * www
ip filter 200032 pass * 106.185.156.3 tcpflag=0x0002/0x0fff * 21
ip filter 200039 reject * *
ip filter 200099 pass * * * * *
ip filter 500000 restrict * * * * *
ip filter dynamic 200080 * * ftp
ip filter dynamic 200081 * * domain
ip filter dynamic 200082 * * www
ip filter dynamic 200083 * * smtp
ip filter dynamic 200084 * * pop3
ip filter dynamic 200085 * * submission
ip filter dynamic 200098 * * tcp
ip filter dynamic 200099 * * udp
ip filter dynamic 201 * 106.185.156.2 www
ip filter dynamic 202 * 106.185.156.3 ftp

# 【IPv6フィルタ設定】
ipv6 lan2 secure filter in 200030 200031 200038 200039
ipv6 lan2 secure filter out 200099 dynamic 200080 200081 200082 200083 200084 200085 200098 200099
ipv6 filter 200030 pass * * icmp6 * *
ipv6 filter 200031 pass * * 4
ipv6 filter 200038 pass * * udp * 546
ipv6 filter 200039 reject * *
ipv6 filter 200099 pass * * * * *
ipv6 filter dynamic 200080 * * ftp
ipv6 filter dynamic 200081 * * domain
ipv6 filter dynamic 200082 * * www
ipv6 filter dynamic 200083 * * smtp
ipv6 filter dynamic 200084 * * pop3
ipv6 filter dynamic 200085 * * submission
ipv6 filter dynamic 200098 * * tcp
ipv6 filter dynamic 200099 * * udp

# 【telnet設定】
telnetd host lan1

# 【DNSの設定】
dns host lan1
dns service fallback on
dns server dhcp lan2
dns server select 500000 dhcp lan2 any .
dns private address spoof on

(続く)

```

■補足

- ・赤字は確定通知書に記載されている情報から、設定しています。
- ・フィルタ設定は必要に応じて適宜変更を行ってください。
- ・LAN1接続機器(サーバ)がIPv6を使用せずIPv4通信のみを行う場合、以下は設定しないでください。

```

ipv6 prefix 1 ra-prefix@lan2::/64
ipv6 lan1 rtadv send 1 o_flag=on
ipv6 lan1 dhcp service server

```

4. 設定について(2/2)

```
# 【IPv6アドレスの通知設定】
ipv6 lan1 prefix change log on
lan linkup send-wait-time lan2 5
schedule at 1 startup * lua emfs:/v6plus_address_notification.lua

# 【Luaスクリプトの設定】
embedded file v6plus_address_notification.lua <<EOF
-----
-- 設定値                                     --
-----
UPD_SV = "http://fcs.enabler.ne.jp/update"
USERNAME = "N123456789"
PASSWORD = "CAF1234567890"
IPv6_IF = "LAN1"
LOG_PTN = "Add%s+IPv6%s+prefix.+(Lifetime:%s+%d+%s+via%s+" .. IPv6_IF .. "%s+by"
RETRY_INTVL = 10
RETRY_NUM = 3
LOG_LEVEL = "info"
LOG_PFX = "[v6plus]"
FAIL_MSG = "アップデートサーバーへの通知に失敗しました。(リトライ: 残り%d回)"
-----
-- SYSLOGを出力する関数                       --
-----
function logger(msg)
    rt.syslog(LOG_LEVEL, string.format("%s %s", LOG_PFX, msg))
end
-----
-- メインルーチン                             --
-----
local rtn, count, log, result
local req_t = {}
local res_t

req_t.url = string.format("%s?user=%s&pass=%s", UPD_SV, USERNAME, PASSWORD)
req_t.method = "GET"

while true do
    rtn = rt.syslogwatch(LOG_PTN)

    if rtn then
        count = RETRY_NUM

        while true do
            res_t = rt.httprequest(req_t)

            if res_t.rtn1 then
                logger("アップデートサーバーへの通知を実行しました。")

                if res_t.code == 200 then
                    result = "成功"
                else
                    result = "失敗"
                end

                log = string.format("アップデートに%sしました。(code=%d, body=%s)", result, res_t.code, res_t.body)
                logger(log)
                break
            end

            -- リクエストに失敗
            count = count - 1
            if count > 0 then
                -- (RETRY_INTVL)秒後、リトライ
                logger(string.format(FAIL_MSG, count))
                rt.sleep(RETRY_INTVL)
            else
                -- リトライ上限を超えたので諦め、SYSLOGの監視を再開する
                logger("アップデートサーバーへの通知に失敗しました。")
                break
            end
        end
    end
end
end
EOF
```