第 IX 部

Linux における IPv6/IPsec スタックの研究開発

第 9 部
Linux における IPv6/IPsec スタックの研究開発

第 1 章　USAGI Project の概要と目的

USAGI Project は、Linux を中心としてより良い IPv6 環境を提供することを目的に、有志によって構成されたプロジェクトである。WIDE Project、KAME Project、TAHI Project と連携をとりながら、Linux の IPv6 スタックや、IPv6 に関するライブラリ、アプリケーションを改良し、より良いコードを提供している。それらの成果物は、2 週に 1 度の snapshot と、年数回を目処とした stable release として公開している。プロジェクトに関する最新の詳しい情報についても http://www.linux-ipv6.org にて公開している。

現在、USAGI Project はメインラインカーネルに対して、USAGI kernel パッチの送付を行っている最中である。多くの改善点と機能は既に送付し、採り入れられている。また USAGI Project はこれからもパッチの送付を続けていく。

USAGI Project の目標は、メインラインカーネルへの完全なるマージである。

第 2 章　2003 年の主な活動

2.1 IPsec

（研究項目）

Linux IPsec の設計と開発

（背景）

これまで、Linux には、IPv6 をサポートした IPsec の実装は存在しなかった。このため、USAGI Project では、2001 年、2002 年と独自に IPv4、IPv6 の両方をサポートした IPsec の実装を行ってきた。この実装を、2002 年 9 月に Linux Network 開発者メーリングリストである netdev に提出したが、メンテナは、異なる設計思想のアーキテクチャを持つ IPsec の実装の導入を準備していたため採用されなかった。しかしこの実装は IPv4 のみをサポートし IPv6 に対する実装は含んでいなかった。このため USAGI Project では、新たに導入されたアーキテクチャに基づく IPv6 の実装を行うこととなった。

（実現されている機能）

現在 Linux では、IPv4、IPv6 をサポートした IPsec が利用可能である。サポートされている IPsec プロトコルは、AH（Authentication Header）、ESP（Encapsulated Security Payload）および IP Compression である。サポートされているアルゴリズムは、認証アルゴリズムとして MD5、SHA1、SHA256、RIPEMD160、暗号アルゴリズムとして DES-CBC、3DES-CBC、CAST128-CBC、BLOWFISH-CBC、AES-CBC、SERPENT-CBC、TWOFISH-CBC、NULL 暗号、Compression アルゴリズムとして deflate である。ユーザランドの鍵交換デーモンは、KAME Project の racoon を使用するが、http://ipsec-tools.sourceforge.net/から Linux 上でコンパイル可能なコードを入手することができる。

（今までの成果とステータス）

すべて、新たなアーキテクチャに基づく。

2003/1　PF_KEY Interface の実装
2003/2　IPv6 IPsec サポートの実装
2003/3　IPv6 IPsec サポートパッチのポスト
2003/3　IPv6 IPsec を Linux がサポート
2003/3　IPsec の処理部分導入にともない他の IPv6 拡張ヘッダの扱いを変更
2003/4　IPv6 IPsec に関連した変更として append data アーキテクチャを実装
2003/4　IPv6 の出力処理が append data へ移行
2003/5 以降　IPv6 IPsec のバグフィックス
2003/10　通常パケット（TCP、UDP など）の出力ルーチンと Neighbor Discovery パケット出力ルーチンの統合

（コメント）

過去に USAGI Project が Linux-2.4 に対して実装した IPsec サポートコードは、メンテナンスの関係上、bHISTORIC-IPSEC-MIP6-20030804 ブランチに移行し、基本的にはすでにメンテナンスを終了した。今後は、Linux-2.6 で導入されたアーキテクチャに基づく IPsec を Linux-2.4 にバックポートする予定である。なお、鍵交換デーモンに関しては WIDE/IPsec WG でサポートしていくことになった。

2.2 Mobile IPv6

（研究項目）

Linux IPv6 Mobility の設計と開発

（背景）

USAGI Project では、ヘルシンキ工科大学（HUT）で開発された Mobile IPv6 プロトコルスタックの実装を元に、IPsec との協調処理などの USAGI Project 独自の拡張機能を実装してきた。

2002 年 10 月に HUT は、2.5 系カーネル向けに Mobile IPv6 プロトコルスタック実装のパッチをカーネルメンテナに提供したが、Mobile IPv6 に特化したカーネルの修正が多く、また、Mobile IPv6 の仕様自体が標準化されていなかったため、カーネルに取りこまれることに対して理解を得られず、このパッチは受け入れられなかった。そこで USAGI Project では、2.5 および 2.6 系カーネルで実装されている既存の枠組み（IPsec で利用されている XFRM）を流用しつつ、カーネル内で持つ必要のないデータ構造や処理の一部はカーネルと分離し、ユーザランドで実装するデザインを行い、実装することにした。

さらに、HUT との共同作業を開始し、作業の効率化を図り、Mobile IPv6 プロトコルスタックが Linux の機能として取り入れられることを目標としている。

（実現される機能）

現在 draft-ietf-mobileip-ipv6-24.txt として公開されている Mobile IPv6 プロトコルスタックが利用可能となる。

（今までの成果とステータス）

2003/3　USAGI Project 内部で XFRM を利用したカーネルとユーザランドによる Mobile IPv6 プロトコルスタックの実装方針の設計

2003/10　HUT との共同作業開始、カーネルとユーザランドとの機能分離についての設計

（コメント）

過去に USAGI が Linux-2.4 に対して実装した Mobile IPv6 サポートコードは、bHISTORIC-IPSEC-MIP6-20030804 ブランチとして保持されているが、メンテナンスは終了している。

ただし、SHARP 製 PDA の Zaurus では、現在 2.5 および 2.6 系カーネルに対応していないため、Zaurus 用に 2.4 系カーネルのバックポートは行っている（次節参照）。

2.3 zaurus

（研究項目）

Zaurus を用いた Mobile IPv6 の実証実験

（背景）

USAGI Project では、IPv6 の普及を促進する活動の 1 つとして、PDA へ USAGI カーネルを搭載し、動作検証を行っている。特に、PDA の特性から、モビリティ技術（Mobile IPv6 プロトコルスタック）を中心に検証を進めてきた。

PDA は、シャープ株式会社から発売されている Zaurus SLC700、SLC750、SLC760 を用い、USAGI プロジェクトで開発が進められている 2.4 系のカーネルの新機能を Zaurus が標準搭載するカーネルバージョン（2.4.18）へ逐次バックポートする手法で、USAGI IPv6 スタックを搭載している。

さらに、SIP クライアント（Linphone）や動画再生ソフト（mplayer）などのアプリケーションを Zaurus 上で動作させ、より生活シーンを意識した環境での検証を進めることで、IPv6 の有用性をアピールすることを目的に活動している。

（実現される機能）

現在 draft-ietf-mobileip-ipv6-24.txt として公開されている Mobile IPv6 プロトコルスタックが PDA 上で利用可能となる。

（今までの成果とステータス）

2003/7　NETWORLD+INTEROP 2003 TOKYO IPv6 ShowCase へ出展。ZaurusSLC700 を Mobile IPv6 の移動ノードとして、CISCO の

ホームエージェントと接続し、動画再生のデモンストレーションを行った。

2003/9　ETSI 4th IPv6 Plugtest 参加。ZaurusSLC760 を Mobile IPv6 の移動ノードとして、Mobile IPv6 ID24 の相互接続試験を行った。

2003/12　Nautilus Project L3 mobility 実験への提供。ZaurusSLC760 を Mobile IPv6 の移動ノードの機能および SIP クライアント等のアプリケーションを搭載して提供し、Mobile IPv6 プロトコルスタックの実験を行っている。

（コメント）

Mobile IPv6 プロトコルスタックおよび IPsec の開発は 2.6 系を中心に進められており、今後、Zaurus の Mobile IPv6 および IPsec スタックは、2.6 系で開発されたものをバックポートしていく予定である。

2.4 netfilter

（研究項目）

Linux IPv6 Connection Tracking の実装

（背景）

Connection Tracking は、IPv4、IPv6 スタックに入ってくるパケット全てを解析し、コネクションの状態の変化を追跡する機能である。この機能は既に Linux の IPv4 スタックに存在していたが、IPv6 スタックには存在せず、長らく実現が望まれていた。

（実現される機能）

Connection Tracking を利用して、例えば Stateful Packet Inspection（SPI）を容易に実現できる。SPI は、コネクションの状態を用いたフィルタリングルールである。

例えば SPI をホストに適用すれば、自分のホストが Initiator となるコネクションのパケットを通過させる一方で、他のホストが Initiator となるコネクションのパケットを全て破棄することができる。

（今までの成果とステータス）

2003/7　IPv6 版 Connection Tracking の実装開始

2003/8　IPv6 版 Connection Tracking のコーディング完了。テスト開始

2003/9　IPv6 版 Connection Tracking を Linux Kernel Mailing List、Netfilter Project（Linux のパケットフィルタ開発を担当しているプロジェクト）の Mailing List へ投稿

2003/10　IPv4、IPv6 に非依存な Connection Tracking の実装開始

2003/12　IPv4、IPv6 に非依存な Connection Tracking 第 1 版の実装完了

（コメント）

9 月、Linux Kernel Mailing List に投稿した IPv6 版の Connection Tracking は、IPv4 版とコードが似ており、統合すべきとの意見が多く、メインストリームのカーネルには採用されなかった。そこで、IPv4 版、IPv6 版を統合した、レイヤ 3 プロトコルに非依存な Connection Tracking を開発している。

付録 A　リリース履歴

[snapshot release]

2003/12/22　usagi-24, usagi-26
2003/12/08　usagi-24, usagi-26
2003/11/24　usagi-24, usagi-26
2003/11/10　usagi-24, usagi-26
2003/10/28　usagi-24, usagi-26
2003/10/13　usagi-24, usagi-26
2003/09/29　usagi-24, usagi-26
2003/09/15　usagi-24, usagi-26
2003/09/01　usagi-24, usagi-26
2003/08/14　usagi-24, usagi-26
2003/08/04　usagi-24, usagi-26
2003/07/21　usagi-24, usagi-26
2003/07/07　usagi-24
2003/06/23　usagi-24
2003/06/09　usagi-24
2003/05/26　usagi-24
2003/05/12　usagi-24
2003/04/28　usagi-24
2003/04/14　usagi-24
2003/03/31　usagi-24
2003/03/21　usagi-24

2003/03/03　usagi-24
2003/02/17　usagi-24
2003/02/03　usagi-24
2003/01/20　usagi-24
2003/01/06　usagi-24

[stable release]

2003/02/14　USAGI STABLE RELEASE 4.1

付録 B　PATCH リスト

USAGI プロジェクトの成果はすでに linux kernel の mainline に数多く取り込まれている。

2003 年中に ML に submit したパッチのリストを以下に添付する。

- ML での議論により revise して再 submit したパッチは重複して記述した。
- 小さな修正に対する patch も 1 つ（e-mail 1 通につき 1 patch）として数えた。

1/05　[PATCH] IPv6: Fix Length of Authentication Extension Header
1/07　[PATCH] IPsec Configuration Extension for IPv6
2/19　[PATCH] IPv6 IPsec support
2/20　[PATCH] dst->{in,out}put() clean-up
2/22　[PATCH] IPv6 IPSEC support
2/28　[PATCH] Use C99 initializers in net/ipv6
3/05　[PATCH] IPv6 IPsec support
3/12　[PATCH] IPSEC: typo in xfrm_sk_clone_policy()
3/19　[PATCH] IPv6 Extension headers (Re: [PATCH] IPv6 IPsec support)
3/23　[PATCH] IPv6: use “const” qualifier
3/23　[PATCH] IPv6: use RFC2553 constant
3/23　[PATCH] IPv6: use ipv6_addr_any() for testing unspecified address
3/30　[PATCH] IPv6: Don’t assign a same IPv6 address on a same interface (is Re: IPv6 duplicate address bugfix)
4/09　[PATCH] MOD_{INC,SEC}_USE_COUNT() in net/ipv{4,6}
4/15　[PATCH] [NET] use fl6_{src,dst} etc.
4/15　[PATCH] [IPV6] Fixed multiple mistake extension header handling
4/17　[PATCH][IPV6] Introduce ip6_append_data
5/08　[PATCH] IPv4 IPComp
5/16　[PATCH] IPv6 IPComp
5/18　[PATCH] IPV4 IPComp: threshold comparison
6/01　[PATCH] xfrm ip6ip6
6/04　[PATCH] IPV6: Sereral errors on udpv6_connect()
6/04　[PATCH] IPV6: typo, unrequired #undef and killing warning
6/06　[PATCH][IPV6] keeping dst refcnt correctly with using xfrm
6/07　[PATCH] fix esp6 extension headers handling
6/11　[PATCH/RFC] IPV6: Remember Manage/OtherConfig flags
6/12　[PATCH] IPV6: fix payload length of reassembled packet
6/12　[PATCH] IPV6: eliminating magic number for sizeof(struct frag_hdr) (Re: [PATCH] IPV6: fix payload length of reassembled packet)
6/14　[PATCH] xfrm ip6ip6 (revised)
6/14　[PATCH] [XFRM] xfrm_alloc_spi() always selected minspi
6/15　[PATCH][IPV6] fix ipv6 header handling of AH input.
6/20　[PATCH] [IPV6] clean-up advmss calculation
6/24　[PATCH] IPV6: use macro for maximum payload length
6/24　[PATCH] IPV6: Fix large packet length check
6/26　[PATCH] IPV6: DAD has to be destined to solicited node mulitcastaddress
6/26　[PATCH] IPV6: DAD must not have source link-layer option
6/26　[PATCH] IPV6: inappropriate static variable in net/ipv6/ndisc.c
6/26　[PATCH] IPV6: Fixed fragment check in ip6_output.c:ip6_fragment()
6/28　[PATCH] IPV6: Fixed M-Flag in last fragment

6/28 [PATCH] IPV6: use macro for M-Flag and clean-up
6/29 [PATCH] IPV6: convert /proc/net/ip6_flowlabel to seq_file
6/29 [PATCH] XFRM: typo
6/30 [PATCH] IPV{4,6}: fixed /proc/net/raw{,6} seq_file support
7/01 [PATCH] IPV4: convert /proc/net/igmp to seq_file
7/01 [PATCH] IPV4: convert /proc/net/mcfilter to seq_file
7/01 [PATCH] IPV6: convert /proc/net/igmp6 to seq_file
7/01 [PATCH] IPV6: convert /proc/net/mcfilter6 to seq_file
7/01 [PATCH] IPV6: convert /proc/net/anycast6 to seq_file
7/01 [PATCH] seq_file conversion /proc/net/igmp
7/03 [PATCH] IPV6: fix a mistake in ipv6_advmss() conversion
7/03 [PATCH] NET: fix SEGV/OOPS with /proc/net/{raw,igmp,...} (is Re:[Bug 863] New: cat /proc/buddyinfo + netstat -a kills machine)
7/04 [PATCH] NET: disconnect support by null address (is Re:Disconnecting a connected UDP socket)
7/04 [PATCH] IPV6: remove unused variable
7/08 [PATCH] IPV6: Fix BUG when appending destination options headers
8/06 [PATCH][IPV6] fix clearing in ah6 input
8/06 [PATCH][IPV6] fixed authentication error with TCP
8/08 [PATCH] IPV6: strategy hander for net.ipv6.conf.∗.forwarding (is Re: problem setting net.ipvX.conf.all.forwarding via sysctl() system call)
8/08 [PATCH] IPV6: typo in include/linux/ipv6.h
8/09 [PATCH] IPVS: linkage error without CONFIG_IP_VS_PROTO_TCP
8/16 [PATCH] SCTP: typo in Kconfig
9/06 [PATCH] /proc/net/{igmp,msfilter,raw,rt_cache,ip6_flowlabel,msfilter6,raw6} may drop some data (Re: /proc/net/* read drops data)
9/06 [PATCH] /proc/net/if_inet6 may drop some data (Re: /proc/net/∗ read drops data)
9/06 [PATCH] clean up /proc/net/{anycast6, igmp6} (Re: /proc/net/∗ read drops data)
9/08 [PATCH] NET: Use proc_net_fops_create() and proc_net_remove() in net/core
9/08 [PATCH] NET: Use proc_net_fops_create() and proc_net_remove() in net/ipv4
9/08 [PATCH] NET: Use proc_net_fops_create() and proc_net_remove() in net/ipv6
9/10 [PATCH] /proc/net/{igmp,msfilter,raw,rt_cache,ip6_flowlabel,msfilter6,raw6} may drop some data (Re: /proc/net/∗ read drops data)
9/10 [PATCH] /proc/net/if_inet6 may drop some data (Re: /proc/net/* read drops data)
9/10 [PATCH] clean up /proc/net/{anycast6, igmp6} (Re: /proc/net/∗ read drops data)
9/13 [PATCH][IPV4] convert proc/net/pnp to seq_file.
9/13 [PATCH] NET: use proc_net_fops_create() for /proc/net/wireless
9/25 [PATCH]: IPv6 Connection Tracking
10/24 [PATCH] [IPv6][IPsec] fix oops with using IPsec
10/26 [PATCH] IPv6: Fix odd IPv6 header in UDPv6 packets when sending MSG_MORE flag
10/26 [PATCH] NET: store cork'ing flow information in common storage
10/26 [PATCH] IPv6: breakage of sendmsg to IPv4-mapped address via UDPv6 socket
10/27 [PATCH] IPV6: Typo in address comparison
10/27 [PATCH] IPV6: inappropriate usage of inet{,6}_sk()
10/28 [PATCH] IPV4: CONFIG_INET description
10/29 [PATCH] IPV6: CONFIG_IPV6 Documentation update
10/29 [PATCH] IPV6: CONFIG_IPV6 Documentation update

10/29 [PATCH] DECNET: Compaq ⇒ HP

10/29 [PATCH] remove historic entries in Documentation/Changes

11/01 [PATCH] IPV{4,6}: Fix one more inappropriate use of inet6_sk()->ipv6only

11/10 [PATCH] linux/times.h needs asm/param.h

11/10 [PATCH] [IPV{4,6}] Normalize jiffies values reported to userspace

11/11 [PATCH] NET: Normalize jiffies reported to userspace, in neighbor management code

11/11 [PATCH] DECNET: Normalize jiffies reported to userspace

11/11 [PATCH] IPV{4,6}: Normalize jiffies reported to userspace in routing code (missing pieces)

11/25 [PATCH] IPV6: redo stateless addrconf properly