TACA WG 2004 年度活動報告

石山 政浩[1] masahiro@isl.rdc.toshiba.co.jp
井上 淳[2] inoue@isl.rdc.toshiba.co.jp
岡部 宣夫[3] nov@tahi.org
坂根 昌一[4] sakane@kame.net
鎌田 健一[5] kamada@nanohz.org

2005 年 1 月 27 日

[1] 株式会社東芝
[2] 株式会社東芝
[3] 横河電機株式会社
[4] 横河電機株式会社
[5] 横河電機株式会社

第1章　はじめに

TACA ワーキンググループでは、PC のように豊富なリソースを持っていないデバイスがネットワークに接続される環境を想定し、それを可能とする技術の研究や開発を行っている。扱っている問題の例を挙げると、大量のデバイスをどのように運用・管理するかや、計算能力の低いデバイスでいかにしてセキュリティーを確保するかなどである。

本年度は主に制御系ネットワーク (Process Automation, Factory Automation, Building Automation, ...) に IP 技術を取り入れるにあたり直面する課題について研究を行った。具体的には、センサなどに代表される非常に多数で処理能力の低いノードを、いかに安全かつ低コストに設定するかについて研究・開発を行った。また、そのような場面で dependability を確保するために必要な技術について、議論を行なった。

第2章　制御ネットワークのIP化におけるノードの安全な自律設定システムの一検討

2.1　概要

制御ネットワークのIP化における課題の一つとして，センサなどに代表される非常に多数で処理能力の低いノードの設定を，いかに低いコストで，かつ安全に行なうかという問題がある．本章では，Kerberosと，鍵交換プロトコルKINKを用いてセキュリティを提供し，ノードの設定情報を管理するサーバと，これらのサーバをノードが自律的に発見できる機構を提案する．提案方式は，ノードが起動時に必要な情報はノードのIDとKerberosのための鍵のみであり，他の設定情報は自律的に発見，設定するため，管理コストが低く安全なノードの設定システムとなることを示す．

2.2　はじめに

近年，ビルに代表されるさまざまな建築物に対してネットワークを利用した高機能化へのニーズが高まっている．例えばビルにおいては，照明や空調などの制御をネットワークを利用して緻密にコントロールすることにより，建物全体のエネルギー消費量を下げることができる．これはビル全体のライフサイクルコストを低減し，環境への配慮のみならず，ビル自身が生み出す収益の向上につながる． また，ビルのユーザに対してさまざまなサービスをネットワークを介して提供することができる．例えばセキュリティシステムとエレベータシステムをネットワークで連係させ，テナントがすべてクローズしたフロアには一般のエレベータを止めないことでフロア全体の安全性を強化できるといったことがある．これらは顧客の満足度を高め，ビルの価値を高める．このように，ビルのネットワーク利用によるインテリジェント化はさまざまなアプリケーションがあり，多くのビルがさらなるネットワーク化に取り組んでいる． 一方で，ビルなどで利用される制御機器のネットワーク化はすでに始まっており，LonWorks[5], BACnet[2], EMIT[1]などの制御ネットワークの仕様の標準化も進んでいる．

しかし，現在はさまざまな機器と制御対象を接続したいという要求が強い．例えば自分の携帯電話から直接部屋のエアコンの制御を行なう，あるいは警備会社のシステムとフロアのカメラを接続してモニタを行うなど，さまざまなシステムを連係動作させて新たな付加価値を生み出そうとしている．このような状況においては，制御ネットワークは専用の

プロトコルではなく，現在最も利用されているネットワークプロトコル，すなわち Internet Protocol (IP) の利用が求められている． IP の利用は，既存のネットワークインフラストラクチャを利用した高度な応用が期待できる．また，現在は管理システム側の IP 化が進んでおり，IP 化された制御ネットワークがインターネットと接続しない場合においても，管理システムとのプラットホームの共通化によるコスト減が考えられる． 同様に，運用ノウハウの共通化によるソフト的なコストも軽減されうる．加えて，ハードウェアの観点においても，トランシーバなどネットワークの物理層のハードの共有化による全体コストの削減も期待できるため，制御ネットワークの IP 化への要求は高い． さらに，多数のセンサーやアクチュエータがネットワークに接続されることが予想されるため，制御ネットワークの IPv6 化が強く望まれている．

だが制御ネットワークの IP 化にはまださまざまな課題が残されている．まず，制御ネットワークには非常に多数のノードが接続されることが考えられる．センサーやコントローラなどが IP 化された場合，ネットワーク管理者は現在よりも遥かに多数のノードを管理する必要が生じる．前述のようにコストの削減も IP 化への一つの理由である以上，低いコストでこれら多数のノードを設定，管理しなければならない．セキュリティも重要な課題である．従来は制御ネットワークは仕様も公開されていない場合があり，一般のユーザが制御ネットワークへ接続するのは難しかった．しかし IP 化されると，一般のユーザが誤って接続してしまうこともありうる．また，悪意を持ったユーザが制御ネットワークへの攻撃することも従来に比べれば容易になる．

本章では，この問題に着目し，制御ネットワークのような非常に多数のノードが IPv6 を利用して接続される環境において，管理者が少ない設定コストで，各ノードを安全に設定，管理できる手法を提案する．提案方式では，Kerberos[7] を用い，Property Server と呼ばれる各ノードの設定情報を保持するサーバを導入し，ノードがこれらを自律的に発見することで，管理者は簡易かつ安全に多数のノードが設定可能となる．

2.3 提案方式

2.3.1 要求条件

本章では，制御ネットワークにおいて，多数のノードを接続した場合においても，管理者が初期設定情報を容易に，かつ安全に設定できることを可能とするシステムを提案する．ここで言うノードの設定情報とは，たとえばノードが照明のスイッチであった場合，どの照明装置に対してオン/オフを伝えれば良いかなどの情報や，そのスイッチが物理的にどこに設置されたかなど，ノードに纏わる情報を指す．また，ノードが通知する，ノードの現在の状態になどに関する情報も含まれる．これにはたとえば現在の IP アドレスなどが挙げられる．多数のノードを設定する場合に，管理者は個々のノード毎に設定するよりも，ネットワークのどこかにノードの設定情報を集約しておき，各ノードが自律的に自己の設定情報を取得して起動するほうが，特に多数のノードが存在する場合には総合的なコストが低くなる． 一方で，ノードがネットワークを介して設定を行なう場合，なりすましや盗聴に対する防御も必要である．

これらのことを踏まえ，まず制御系ネットワークのIP化に対する要求条件を以下に示す．

1. 低い管理コスト
   制御ネットワークは通常専任の管理者が想定可能であるため，家庭用ネットワークのような完全な無設定となるシステムを目指す必要はない． しかし，ノード数が膨大であるため，個々のノード単位での設定は必要最小限にとどめ，それぞれのノードが可能な限りネットワークを通して自律的に設定できるアーキテクチャが望まれる．

2. セキュリティ
   既存の公開されたネットワーク技術を用いるため，制御ネットワークでの盗聴などのさまざまな攻撃が想定される．このため，すべてのノードがセキュリティを確保するプロトコルを利用できるようにする枠組が必要がある．

3. ノード数に対するスケーラビリティ
   前述のように制御ネットワークに参加するノードは非常に多くなることが想定されるため，ノード数に対してのスケーラビリティがシステムには求められる．

4. 低コストノードでの運用可能性
   一般に部品コストの低いノードは計算能力も低く，特に公開鍵暗号系のような多倍長整数演算を必要とするような処理を行なうことは現実的でない場合が多い．このようなノードが参加可能で，かつ全体のセキュリティがこのような低コストノードによって下がらないシステムが必要となる．

これらの要求条件を踏まえ，われわれは制御ネットワークのIP化におけるノードの自律設定をサポートし，大規模なネットワークシステムを運用管理するための基盤となるシステムを提案する．

2.3.2 提案方式のシステム構成

提案システムの概要を図2.1に示す．提案システムはKerberosをベースとしたシステムである．KDCはKerberosのKey Distribution Centerであり，管理対象となるすべてのノードと鍵を共有する．また，すべての管理対象ノードはKerberosのclientである．

管理対象となるノード (たとえば各空調装置や照明装置など) はKDCと共有される鍵を持っていることを前提とする．また，自ノードの識別子を持っているものとする．ここでは一般的なネットワークデバイスを想定し，すべてのノードはEUI-64のアドレスを付与されていると仮定する． 加えて，すべてのノードはIPsec[6]を利用できるものとし，IPsecの鍵交換プロトコルはKINKを使用できるものとする．

Property Server (PS) は管理対象ノードの設定情報や属性を保持するノードサーバである．PSは管理対象ノードと同一のネットワークにあってもよいし，またインターネット上にあってもよい．PSもまた管理対象ノードと同様に，Kerberos Clientであり，KDCと鍵を共有する．また，IPsecを利用できるものとし，IPsecの鍵交換プロトコルはKerberosを利用した鍵交換方式であるKINKを使用できるものとする．

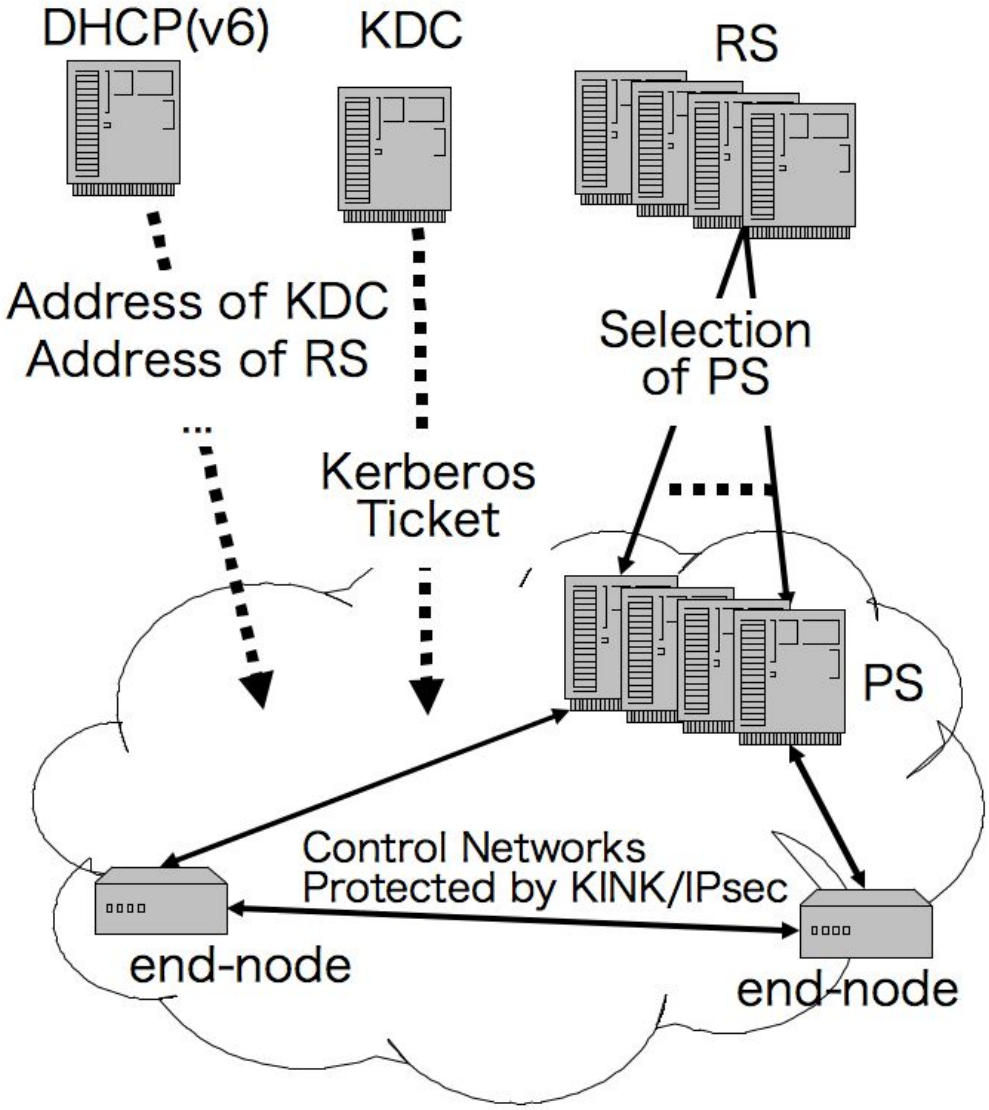

図 2.1: 提案システムの概要: すべてのノードは Kerberos の管理下におかれる．各ノードは設定情報を保持する Property Server を Rendezvous Server を通して発見し，自律的に設定情報を安全に取得する．

提案方式では，あるノードの設定情報をどの PS が持っているかを保持するデータベースを持つ．これを Rendezvous Server (RS) と呼ぶ． 本提案方式では RS としてローカルな DNS[8] サーバを利用する．この DNS サーバは基本的にグローバルなインターネット上の DNS 構成木とは切り離されて管理される．

また，提案システム内には，KDC および RS のアドレスを提供するため，DHCPv6[4] サーバがあるものとする．DHCPv6 には, Kerberos に必要な情報を付与するための拡張が行なわれているものとする．この拡張には，例えば対象となる KDC が管理する Realm 名の通知などが挙げられる．

2.3.3 通信モデル

以下図 2.2 を例に．ノードの起動時の通信手順を示す.

D1 は DHCPv6 サーバ，K1 は KDC, RS1 は RS を表す．PS1 は N1 の PS とする． また，realm 名を foo.com, N1 の PS の FQDN を ps1.foo.com とする． N1 の EUI-64 アドレスを 0123:4567:89ab:cdef とする．Kerberos では KDC とノードの時刻同期が必要であるが，文献 [3] に従うものとする．

1. ノードの principal 名の決定: ノードは起動後，まず自己の principal 名を決定する．提案方式では，principal 名はノードに与えられた EUI-64 を 16 進数で表記し，4bit ごとに区切ったものとする．この例では，ノード N1 の EUI-64 が 0123:4567:89ab:cdef であるので，N1 の principal 名は 0.1.2.3.4.5.6.7.8.9.a.b.c.d.e.f となる．よってこの principal 名は静的にノードと結び付いていることになる．

2. KDC の発見: N1 はまず DHCPv6 を使用して起動に必要な情報を得る．この情報には，KDC の IP アドレス，ポート番号，NTP サーバのアドレス，RS のアドレス，このネットワークの Realm 名などがある． この例では，realm 名として foo.com が得られる．また，RS のアドレスとして RS1 を得る．

3. PS の発見: N1 は，得られた RS のアドレスのうちの一つに，rev(principal).realm.ps.local の PTR レコードを問い合わせる． ここで，rev(principal) は principal 名を 4bit づつ逆順に並べたものであり，realm は DHCPv6 で得た realm 名である．.ps.local はそのままの文字列である．この例での問い合わせは，f.e.d.c.b.a.9.8.7.6.5.4.3.2.1.0.foo.com.ps.local に対する PTR レコードを RS1 に問い合わせる．

   最後のラベル.local は，誤設定等によって外部の DNS への問い合わせを防止するために付与している．

   ここで N1 は自ノードの PS の FQDN として，ps1.foo.com を得る．なお RS1 は ps1.foo.com の AAAA を解決できるものとする．

4. PS との KINK を利用した鍵交換: N1 は ps1.foo.com に対して KINK を利用して IPsec 通信のための鍵交換を行なう．このとき，ps1.foo.com の名前解決は，RS を利用する．

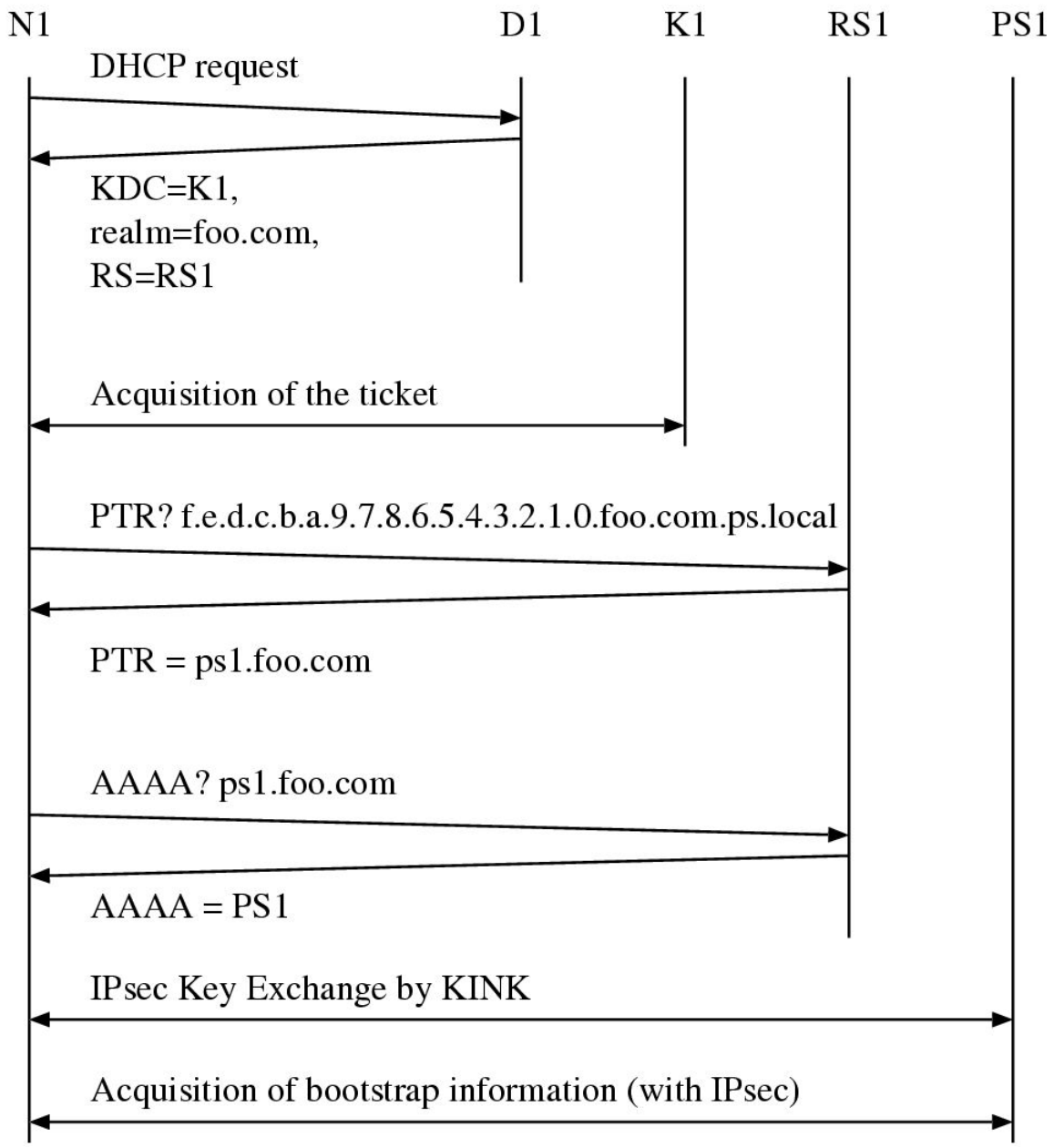

図 2.2: 通信モデル: ノード N1 はまず DHCPv6 を利用して KDC を発見する．自ノードの principal 名を元に，RS を介して PS を発見する．発見した PS から KINK/IPsec によってセキュリティを保持しつつ設定情報を取得する．

5. PS からの設定情報の取得: N1 は IPsec を利用して，PS から自己の設定情報を取得する．

2.4 考察

2.4.1 管理コスト

提案方式では，設定情報を PS に集約することが可能であり，管理者は個々のノードに接続する必要がなく管理コストを低減できる．また，PS は分散配置可能であるため，例えば空調系のノードの PS は建物内に配置し，一方防犯系ノードの PS は警備会社内に配置するといった管理主体の分散も可能である．

また，個々の管理対象ノードには，EUI-64 のような一意な ID と，それに対応する鍵という 2 つの情報のみを擦り込んでおくだけでよい．これはそのノードがどのネットワークで利用されるかに依存しないため，任意の時点，たとえば工場出荷時などに設定可能であり，ノードの生産者にとっても負荷は低い． ノードは接続された時点で自律的に PS を探し出すため，ノードの利用者は KDC に鍵を登録し，PS の設定に専念することができる．よって提案システムの管理コストは低いと言える．

2.4.2 セキュリティ

すべてのノードは KINK を利用した IPsec を利用できるため，PS とノード，あるいは設定後に行なわれる，空調の ON/OFF といったノード間の通信はすべて IPsec が利用可能なため，盗聴やなりすましという攻撃は難しいといえる． 一方，KDC の発見に利用する DHCPv6 においては，攻撃者は容易に DHCPv6 サーバになりすましが可能である．ここで通知されるのは KDC のアドレスであるため，攻撃者は虚偽の KDC を通知することは可能であるが，KDC との共有鍵を知らない限りこの後の通信を成り立たせることは困難である．ゆえに，DHCPv6 のメッセージが正しいかどうかは，与えられた KDC と相互認証可能であるかで判断することができる．

PS を発見するために RS に問い合わせを行なうが，この応答も攻撃者は容易に偽装可能である．しかし，ノードは PS との通信に KINK/IPsec を利用するため，KDC の場合と同様に誤った PS に問い合わせのパケットを投げさせることは可能であるが，その後の通信を成り立たせることはやはり困難である．また必要であれば，RS の通信も KINK/IPsec で保護することも可能である．

RS の発見には DHCPv6 を使用するが，悪意のあるユーザが DHCPv6 サーバになりすまし，誤った KDC および RS の情報をノードに対して与えることができる．しかし，KDC や PS になりすますことは困難であるので，ノードは誤った設定情報であることを検出できる．また，正しい DHCPv6 のパケットを遮断しないかぎり，各ノードは KDC との相互認証が可能になるまで TGT の取得を繰り返すことによって，最終的に正しい KDC と接続できる．加えて，このフェイズはノードが定常状態になる前のブートストラップ時の過渡状態であり，定常状態での性能に影響は出ない．

さらに，ノード間の通信にも KINK を利用した IPsec が利用可能である．よってノード上のアプリケーションも，IPsec が提供するセキュリティ機能が利用可能となる．同様にノード上で実行されるアプリケーションプログラマも，特別なセキュリティメカニズムを意識することなくセキュリティメカニズムを利用できるため，プログラミングのコストを引き上げることがない．

また，もし管理者がノードに事前に擦り込まれた鍵を通信に利用したくないのであれば，擦り込まれた鍵は bootstrap のみに利用し，鍵を新たに設定するという方法も利用可能である．

2.4.3 管理対象ノード数に対するスケーラビリティ

KDC への負荷であるが，基本的に KDC への問い合わせは起動時および，KINK による鍵交換時に発生する．よって，定常運用状態における，各ノードが交換するパケットあるいは張るコネクションに比較すると，KDC への通信量は多くないと考えられる．

一方で，ノードが多数同時に起動するような場合には KDC へのアクセスが集中することが考えられる．KDC が適切に処理要求パケットを破棄すすることによって，KDC 自体の処理を行なうことができる．一方，ノード側は KDC との接続がタイムアウトした場合には充分かつランダムな待ち時間をとって再度 KDC との接続を行なうことにより動作すると考えられる．この手続きによって発生する遅延は定常状態の性能に影響を与えるものではなく，起動時の過渡状態から定常状態への遷移時間へ影響を与えるだけである．さらに，一般的な制御ネットワークにおいては，一ヶ所の電源を投入した時にすべてのノードが起動するような運用は行なわれないのが通常である．

過渡状態から定常状態への移行時間も問題となる場合には，必要な L2 容量の推定と，適切な KDC の分散配置が必要なる．KDC に含まれる情報は静的なものであり，繁雑に更新されることがないため，KDC 自身を多重化することは容易である．しかし管理する KDC の台数が増加するため，この問題は KDC の管理コストと定常状態への移行時間とのトレードオフになると考えられる．

また，管理対象 (ビルや工場や巨大施設) が大きい場合や複雑な場合には管理ポリシーなどを分ける必要が発生する可能性がある．この場合には Kerberos の管理空間，すなわち realm を分割する必要がある．分割した複数の管理空間で相互運用性については，Kerberos の inter-realms を適用することが可能である．

一方で，PS の台数は管理対象の数や特性によって変化するため，柔軟に配置できる必要がある．提案方式では，ノードと PS の関係は，RS に記述するため，PS の台数および位置は自由に設定できる．また RS 自身についても，既存の DNS をそのまま利用可能であるため，ノード数や問い合わせ数に応じて RS の台数を設定できる．

2.4.4 低コストノードでの運用可能性

提案システムは Kerberos を利用したシステムである．IPsec の鍵交換時プロトコルにも，公開鍵系を利用する IKEv2 を利用せず，Kerberos 介して鍵交換を行なう KINK を使用す

る．提案システムはセキュリティメカニズムの中に公開鍵暗号系の方式をまったく必要としないため，計算能力の低いノードにもセキュリティを提供できる．

2.5 おわりに

本章では，多数のノードが接続されると想定される制御ネットワークの IPv6 化において，多数のノードの設定を低いコストで，かつ安全に行なうことを支援するシステムの検討を行なった．提案方式は Kerberos と，Kerberos をベースとした IPsec のための鍵交換プロトコル KINK を用いて，セキュリティを提供し，ノードの設定情報を管理するサーバであるプロパティサーバと，これらをノードが自律的に発見できる機構を提案した．提案方式では，ノードが起動時に必要な情報はノードの ID と Kerberos のための鍵のみであり，他の設定情報は自律的に発見，設定するため管理コストが低い．

今後の課題としては，Property Server における情報の表現方式と，その情報を実際に交換するためのプロトコルの規定などがあげられる．また，提案システムのプロトタイプ実装を行ない，性能評価などを行なっていきたい．

第3章　おわりに

今回は制御系ネットワークにおいて、処理能力の低い大量のノードをいかに設定するかという問題と、その問題に対する自立設定システムの検討結果を報告した。来年度以降は、これらのシステムの試作や性能評価を行う予定である。また、QoS・冗長化・hard real-time 性確保などの要素技術や、運用技術についても議論を行う。

関連図書

[1] *emWare*. http://www.emware.com/.

[2] ANSI/ASHRAE Standard 135-1995. *BACnet–A Data Communication Protocol for Building Automation and Control Networks*. http://www.bacnet.org/.

[3] Don Davis, Daniel Geer, and Theodore Ts'o. *Kerberos With Clocks Adrift: History, Protocols, and Implementation*, January 1996.

[4] R. Droms Ed., J. Bound, B. Volz, T. Lemon, C. Perkins, and M. Carney. *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*, July 2003. RFC 3315.

[5] IEA-709.1. *CONTROL NETWORK PROTOCOL SPECIFICATION*. http://www.lonmark.org/.

[6] S. Kent and R. Atkinson. *Security Architecture for the Internet Protocol*, November 1998. RFC 2401.

[7] J. Kohl and C. Neuman. *The Kerberos Network Authentication Service (V5)*, September 1993. RFC 1510.

[8] P.V. Mockapetris. *Domain names - concepts and facilities*, November 1987. RFC 1034.

Copyright Notice