研究計画書
XMPPを用いたインターネット上での人間と機械の通信の実現

自署:_____

共愛学園前橋国際大学　国際社会学部　国際社会学科

平成22年5月27日

概要　近年における急速な普及の結果、インターネットはコミュニケーションツールとして人々にとって欠かせなものになった。さらに、機械が人間のように振舞う場面も増えてきた。特に、インスタントメッセージングのオープンソースプロトコルであるXMPPにおいては、高い拡張性から機械の状況把握と遠隔操作が実現できるプロトコルとして使うことが出来ると考えられる。しかしながら、現状においてはその実装の際に発生する問題に対して、ソフトウェア的に解決すべき問題というよりは、プロトコル側で解決すべき問題が存在していることが確認されている。本研究ではこれらを踏まえて、その解決方法を提案し、実装を行うことにより、この問題の解決を試みる。また、その他に機械がXMPPでネットワークにつながる際に問題となる事象についても調査し、解決を試みる。

1　はじめに

インターネットが爆発的に普及した結果、人々はインターネット上で様々な活動を行うようになった。これらの人々は、インターネットを離れた人とつながるコミュニケーションツールとして使っている。将来的にはインターネットには人間と機械が、その差を意識すること無く混在して通信している状況が進むと考えられる。これは人間にとって、機械を操作するにあたっても人間に対して通信しているようなインターフェースが扱い易いということから発展していると考えられる。本研究では、インターネット上での人間対機械の通信が今後より一層扱い易くなるよう、人間対人間の通信技術として発展したチャット技術を応用し、人間対機械の通信における問題の解決を試みる。本研究計画書では、第二章で本研究の背景となるインスタントメッセージング技術の代表的なものとしてXMPPについて説明すると共に近年におけるインターネット上の機器の多様化について説明する。第三章では、既存のXMPPの技術を機械に適用した場合における問題について、いくつかの事例を挙げて説明する。第四章では、第三章で挙げた問題について解決方法を挙げる。

2　背景

本章では本研究の背景としてインスタントメッセージング技術、並びに、近年におけるインターネットに接続される機器の多様化について述べる。

2.1　インスタントメッセージング

人と人が文字ベースでインターネット上で会話をするためにインスタントメッセージング(IM)技術が開発された。現在ではこのIMの技術を用いて日々数多くの人が会話を行っている。近年では、IMのプロトコルのうちの一つとしてExtensible Messaging and Presence Protocol (XMPP) [5] [6] [7] [4]が、オープンな規格として規定され広く利用されている。またその拡張もXMPP Extension Protocol (XEP) [1]として公開されており、Multi User Chat (MUC)のように多人数のチャットルームの実現に用いられたり、ビデオチャットを実現するXEPなどが公開されている。また、大規模なものから、小規模な軍事用のものまで、IMの標準として様々なサービスで利用されている。

2.2　機器の多様化

多くの人々がインターネットにつながっている一方でまた、機械もインターネットにつながっている。さらに、今後つながってくるであろう機械も存在し、これからその数は人間を超える勢いで増えることが予想される。また、IPv6の本格的な普及によって、それまでNAT下の環境にあったデバイスもインターネットにつながることにより、今までよりもさらに多様なデバイスが接続されると考えられる。このような状況により、インターネット上での遠隔制御の容易性がこれまで以上に求められると考えられる。以下に今後インターネット上での容易な操作が必要になると考えられる例を挙げる。

- ネットワーク機器
  すでにネットワークにつながっている機械として

挙げられるのは、スイッチ、ルーターなどのネットワーク機器である。現在これらは物理的にコンソールでつなぐか、リモートで 1 台 1 台ログインし、コマンドを使って設定する場合がほとんどである。この方法は、技術を知らない人々にとって敷居が高い。

- 家電
  家電もまたインターネットにつながっている。一部のデジタルビデオレコーダーなどは、外出先から携帯電話などから録画予約がかけられるようになった。しかしながら、現状では多くの家庭にインターネットがつながっているにも関わらず家電はインターネットにつながっていない。これは、開発者にとって実装しやすく、ユーザーにとって使い易い標準のプロトコルがないことにもよると思われる。
- Smart Grid
  第 3 に挙げられるのはスマートグリッドにおける利用である。スマートグリッドは、電力網と IT 技術を組み合わせることにより、現在の電力網をインテリジェント化する試みである。アメリカ合衆国をはじめとして世界中でこの試みが進められていく予定である。現在、スマートメーターと呼ばれる高機能なメーターの設置が始まっており、今後これらのメーターを遠隔で制御して電力の制御が行われる見込みである。また、これがさらに進められることにより、家電の電力の遠隔制御も行われていくと考えられている。また、IETF などの標準化団体がこれの実現に向けた標準のプロトコルの検討を開始している。このスマートグリッドを実現していくにあたって、セキュリティの観点からの懸念があるとされている。XMPP は前述の通りメッセージの交換はもちろん、セキュアなプロトコルとして開発されているため、スマートグリッドにおいて力を発揮すると思われる。しかしながら、現状においてスマートグリッドに適用させる動きはあるものの、標準化の動きは起こっていない。

3 問題点

本章では、本研究で解決を目指す点について述べる。まず、人間と機械の柔軟な通信を実現するにあたり、XMPP という特定の技術から独立した問題について説明し、次に現時点で IM の代表的なプロトコルである XMPP を用いた場合に解決が望まれる点について説明する。

3.1 人間と機械の通信

このように、多くの人が IM を使い、多様な機械がインターネットに繋がる、つながろうとする動きの中で、今後、一般のユーザーにとって使い易い形での、機械を操作する方法、また、機械から情報を得る方法が必要となると考えられる。さらに将来的には人間が相手を機械として認識せずに会話を行う場合も起こってくると予想される。本研究では、「人間とネットワークにやさしい、機械が参加しているネットワークをつくること」を目標とする。本研究により、様々な機械がインターネットにつなぐにあたって、簡単になり、またその遠隔操作も簡単になることを支援したい。また、すでに、GUI で遠隔操作を実現させているものも数多くある。しかしながら、ユーザーインターフェースの標準化の面から考えた場合、汎用性に乏しく、他の機械に応用することが難しい。そのような場合に、機械同士が文字ベースのプロトコルで通信できる枠組みが期待される。

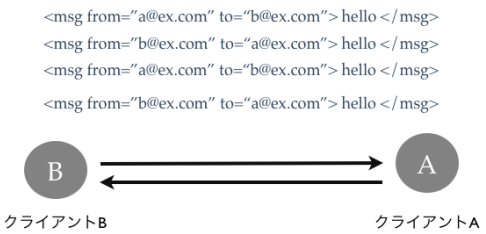

図 1　2 つのクライアント間でループが起こっている場合

XMPP の使用の如何に関わらず、機械同士をネットワークでつなげた場合、ループ、スケーラビリティなどの問題が発生することが予想される。

インターネットにつながる機械は前述の通り数多くある。その中でも最もインターネットに密接につながっている、ネットワーク機器を XMPP のクライアントとして使いたいと考えている。

前述のように、現在のネットワーク機器は、コンソールでつなぐなど、敷居が高い。また、ネットワーク台数が増えた場合ネットワーク管理者にとって負担となっている。さらには、設定のミスなどが監視しづらいことにもつながっている。これらの機器に XMPP のクライアントを走らせ、MUC(Multi-User-Chat) などを使って設定出来るようになれば、同時に何台ものクライアントの設定が出来るようになる。さらに、何人かで同時に設定を行うようにすることも可能となる。これにより、設定をしている様子を監視することが出来るようになりヒューマンエラーを防ぐことが出来るようになったり、設定をしている途中で引き継ぐことも容易になる。

3.2 メッセージのループ

最も起こりやすい単純な問題としてメッセージのループが挙げられる。図 1 のように、同じ組織が作った 2 つのクライアント同士がメッセージのループを起こしてい

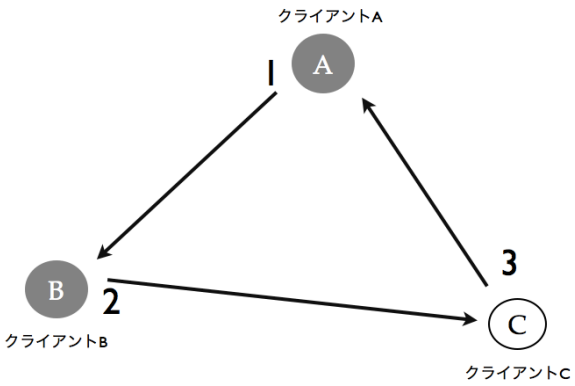

図２　３つのクライアント間でループが起こっている場合

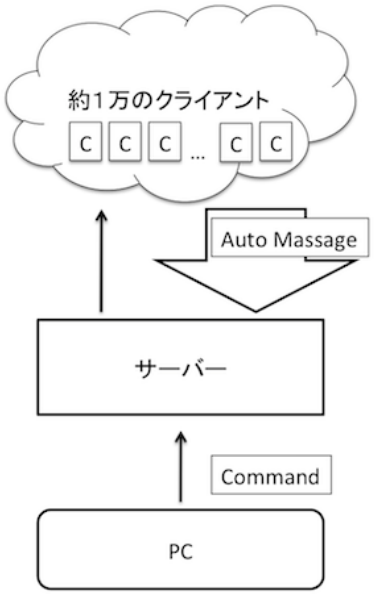

図３　大量のクライアントが一つのサーバーに接続されている場合

る場合を考える。この場合、ソフトウェア作成者がそのXMPP の実装を変更出来るため、プロトコルそのものの問題よりは、ソフトウェアのバグと考えられる。しかしながら、図２のように、A、B、C の３つのクライアントが存在し、A と B が同じ組織が作ったもので、C が違う組織が作ったものであった場合を考える。A は、イベントが起こると、B にメッセージを送信するように設定されている。B は受け取ったメッセージをログとしてディスクに書き込むように設定されているが、書き込めない場合は C にメッセージを送信するように設定されている。C は受け取ったメッセージを A に転送するように設定されている。この場合において、B が C が bot と知らずに、メッセージを送信した場合メッセージのループが起こる。これは単なるバグとは言えない。これはプロトコルのレベルで解決策を検討する必要がある。

3.3　サーバーのスケーラビリティ

例として図３のように１万台規模の大量のスイッチのクライアントがひとつの XMPP サーバーにつながっているような場合において、ユーザーがコマンドを１つのマシンから発すると、小さなメッセージであっても大量のメッセージが届いてしまい、サーバーの CPU、メモリ、帯域が処理を継続できなくなるといった問題が起こる可能性が考えられる。これまでも、サービスが提供できなくなる DoS 攻撃への対処方法は、XEP の形で公開されている。フィルターをかける [3] も考えられているが、これはあらかじめ登録が必要であり、機械のみのネットワークにおいてはあまり現実的でない。[8] によると、サーバーソフトウェアが転送量、stanza の制限を設けることによって、DoS Attack を防ぐことができるとある。しかしながら、この方法だと、ユーザーが欲しい情報が取りだせなくなることになりかねない。[2] は、送信のみ、あるいは受信のみを行うようにすることによって、フィルタリングをする手間を省こうとしているが、この場合結局大量のメッセージは届いてしまうのでほとんど助けにならない。

また、現状では図３の場合において、クライアントが送ったコマンドに対する応答メッセージの中に、他のスイッチにとってコマンドととれる文字列が入っていた場合、そのコマンドに反応してしまう。これもまた、解決方法を検討しなければならない。

このように、すでにわかっている問題もあるが、まだ実態をつかめていない問題もある。本研究では、この他にも機械が XMPP を使う場合にどんな問題が起こるか仮説を立てて、検証し、解決方法を考え、実装して、実際に機能することを確かめる、また動かなかった場合にはそれを問題として捉え、その解決にあたるプロセスを繰り返して研究を進めていく。

4　提案技術

本章では、前述した問題に対する本研究で提案する解決方法について解説する。特に、XMPP を用いて人間と機械が通信するケースにおける具体的な解決策について解説する。

4.1　メッセージのループに対する解決策

ループ問題はその昔から様々なプロトコルで問題となっていたこともあり、その解決方法も様々なものがある。ホップカウント、パスベクトルなどがこれにあたる。

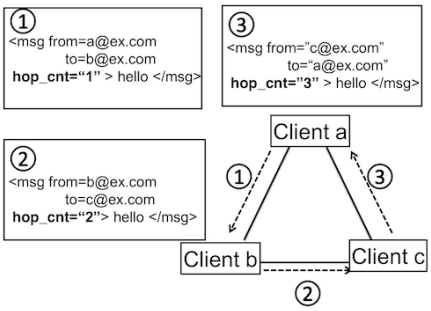

図 4　ホップカウントによるメッセージのループの防止

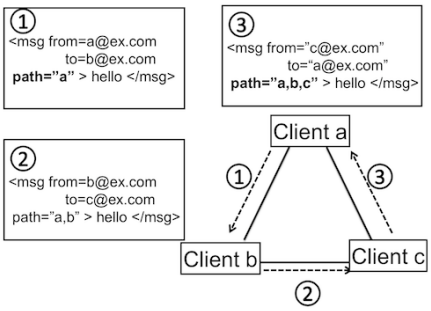

図 5　パスベクトルによるメッセージのループの防止

4.1.1　解決策 1: ホップカウント拡張

解決策としてはまず、TTL Limit(Time To Live Limit) を使う手段である。具体的には、図 4 のように、 stanza に新たに hop_cnt=x というタグを追加しメッセージが通過するたびにこの x の値を増やして行き、ある一定数を超えると自動的にメッセージの転送を止めるような仕組みをつくる方法である。メッセージ自身にホップカウントのフィールドをもたせ、一定数を超えるとサーバーが転送を止める方法である。本研究では本方式による解決策を提案した [9] [10] [11]。

4.1.2　解決策 2: パスベクトル拡張

次に考えられるのは、Path Vector を導入する。方法である。図 5 :XMPP の stanza に新たに path="x" というタグを追加する。メッセージが通過する度に自分の識別子を追加する。メッセージを転送する際にはこの path を読み取り、もし自分の識別子がこの中にあれば転送をやめる仕組みである。本研究では本方式による解決策を提案した [9] [10] [11]。

4.2　スケーラビリティの問題に対する解決策

XMPP は機械でも使えると定義されているものの、1 万クライアントが同時接続して、メッセージを送信といったことは想定されていないと思われる。また、特徴としてランダムなサイズのメッセージではなく、比較的にサイズの似通った小さなメッセージが多い。また、ユーザーが必要な情報を逃さないようにしつつ、サーバーを稼働させ続ける点も重要となる。このため、実現するためにはメッセージの受け取りを拒否する DoS 攻撃からの防御とは違ったアプローチが必要になる。

4.3　スケーラビリティの問題に対する解決策

ここでは前章で述べたスケーラビリティの問題に対する解決策について述べる。

現在、XMPP の拡張としてメッセージの圧縮手法が提案されている [?]。この圧縮手法によって、通信帯域の浪費を防ぐことができる可能性がある。しかしながら、その一方で CPU やメモリなどの計算資源を消費するため、クライアント数が多い場合には CPU やメモリなどの計算資源に対する DoS 攻撃の状態となる可能性も存在する。そのため、サーバ側の計算資源を可能な限り用いずに、圧縮を行うかどうかをクライアント側での独自の計測結果に基づいて決定し、クライアント側で自律分散してサーバの計算量と帯域のトレードオフを調整するなどといった拡張が求められる可能性が高い。特に個々のメッセージが小さい場合には、前述の圧縮は計算資源の浪費につながる可能性も考えられる。

また、現在 XEP-205[?] において静的な設定によって通信量を制限する拡張が提案されている。しかしながら、数万台規模の機器に静的に設定を行うことは現実的には難しいケースが考えられる。そのため、このような場合には、既に多数のネットワーク機器で利用されているバックプレッシャーのような仕組みによって、多数の機器が自律的に調整を行う必要がある。

本研究では、そのような多数の機器がバックプレッシャーなどといった仕組みによって分散して協調的にサーバ側の資源の利用を調整できる拡張も検討する。

5　評価方法

本研究の評価方法としては、どの程度、機械が XMPP のネットワークに参加した場合に問題が起こらないか、また、それに対応した文書が存在するかということになる。

また、本研究では提案する拡張の標準化を目指す。そのため、標準化過程において本研究で提案する解決技術は多数のベンダ等によって充分に評価されると考えられる。

6　期待される成果

本研究は、より機械が XMPP に参加しやすくする。本研究により、人々はより多くの機械を制御しやすくなり、人と機械が現在よりも分かれたものとしてでなく、混ざった状態で生活しやすくなる。例としては、1 万台を超えるネットワーク機器の一括制御、家にある機械を同じ機械、同じソフトウェアからより多く制御することができるようになる、電力の制御をより早く行えるよう

になることで効果的に省電力を可能とすることが出来るようになる。

7 これまでの活動

学部 3 年次より共愛学園前橋国際大学小柏研究室に所属し、ネットワークの基礎を身につけた。学部 3 年次、2009 年 11 月に広島で行われた IETF 76 meeting に参加、2010 年 3 月に機械が XMPP を使った場合に起こる問題についての Internet Draft を提出した。同月にアメリカのアナハイムで行われた IETF 77 Meeting の XMPP Working Group BoF にてプレゼンテーションを行った。以下に一覧を示す。

1. インターネット・ドラフト “Considerations of software generated message on XMPP” draft-sato-xmpp-software-message-00 Internet Draft, 2010
2. インターネット・ドラフト “Considerations of software generated message on XMPP” draft-sato-xmpp-software-message-01 Internet Draft, 2010
3. IETF 発表 “Considerations of software generated message on XMPP”, Hirotaka Sato, XMPP IETF 77, 23 March 2010 - Anaheim, CA, USA, XMPP-WG Session 2010-03-23 1300-1500, March 2010.

8 政策・メディア研究科に進学を希望する理由

政策・メディア研究科では、インターネットのみならず、ガバナンス、イノベーションなどグローバルな視点からの研究が行われている。本研究は、研究をすると共に、継続的な国際貢献を目指している。そのためには、国際的な視点に立っていること、また先進的なインターネット・インフラストラクチャが必要である。また、ネットワークの理論および実践経験を持つ研究指導者が不可欠である。以上の理由から、私は政策・メディア研究科への進学を強く志望する。

9 共同研究者・関連団体

本研究は、共愛学園前橋国際大学 国際社会学部 国際社会学科 小柏研究室において同大学 専任講師 小柏伸夫と共に進めている。また本研究提案者である佐藤弘崇及び小柏は、2010 年 5 月現在、WIDE プロジェクトに参加しており、今後は WIDE プロジェクトにおいても XMPP に関する研究を提案していく予定である。

参考文献

[1] XSF (XMPP Standards Foundation). http://xmpp.org/.

[2] Joe Hildebrand, Jack Moffitt, and Peter Saint-Andre. Stanza Interception and Filtering Technology. XSF (XMPP Standards Foundation) XEP-0273, May 2010.

[3] Peter Millard and Peter Saint-Andre. Privacy Lists. XSF (XMPP Standards Foundation) XEP-0016, February 2007.

[4] P. Saint-Andre and Ed. End-to-End Signing and Object Encryption for the Extensible Messaging and Presence protocol (XMPP). IETF (Internet Endineering Task Force) RFC3923, October 2004.

[5] P. Saint-Andre and Ed. Extensible Messaging and Presence Protocol (XMPP): Core. IETF (Internet Endineering Task Force) RFC3920, October 2004.

[6] P. Saint-Andre and Ed. Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence. IETF (Internet Endineering Task Force) RFC3921, October 2004.

[7] P. Saint-Andre and Ed. Mapping the Extensible Messaging and Presence Protocol (XMPP) to Common Presence and Instant Messaging (CPIM). IETF (Internet Endineering Task Force) RFC3922, October 2004.

[8] Peter Saint-Andre. Best Practices to Discourage Denial of Service Attacks. XSF (XMPP Standards Foundation) XEP-0205, January 2009.

[9] Hirotaka Sato. Considerations of software generated message on XMPP. XMPP IETF 77 - Tuesday 23 March 2010 - Anaheim, CA, USA, XMPP-WG Session 2010-03-23 1300-1500, March 2010.

[10] Hirotaka Sato and Nobuo Ogashiwa. Considerations of software generated message on XMPP. IETF (Internet Engineering Task Force) Internet-Draft (work in progress) draft-sato-xmpp-software-message-00.txt, March 2010.

[11] Hirotaka Sato and Nobuo Ogashiwa. Considerations of software generated message on XMPP. IETF (Internet Engineering Task Force) Internet-Draft (work in progress) draft-sato-xmpp-software-message-01.txt, March 2010.