公衆回線を利用した複数遠隔装置の集中管理方式

HOMMA, Masahiro / 本間, 将紘

(出版者 / Publisher)
法政大学大学院理工学・工学研究科
(雑誌名 / Journal or Publication Title)
法政大学大学院紀要. 理工学・工学研究科編
(巻 / Volume)
56
(開始ページ / Start Page)
1
(終了ページ / End Page)
7
(発行年 / Year)
2015-03-24
(URL)
https://doi.org/10.15002/00011250

公衆回線を利用した 複数遠隔装置の集中管理方式

SIMULTANEOUS MULTIPLE OPERATION METHOD OF REMOTE TERMINAL USING PUBLIC LINE

本間将紘
Masahiro HOMMA
指導教員　金井敦

法政大学大学院理工学研究科応用情報工学専攻修士課程

. Recently, information technology has been exploited in various fields to improve the efficiency of operations. However, in the case of building information gathering system, Maintenance personnel has to go to the building when system trouble occurs because of using a low cost consumer public communication line. It is possible to solve the above problems by using an expensive dedicated line to communicate between control server and clients (building monitoring equipment). However it costs too much. In this paper, worker don't need to go to site because we solve the problem from remote location. We provide the system at a relatively low cost. In this paper, we don't change the client's network to cut the cost. We propose system architecture to realize two way communication between control server and clients and group communication for maintenance using IRC (Internet Relay Chat) concept and low cost consumer public line, furthermore we evaluate the architecture by implementing the prototype system

Key Words *: Remote device, WebSocket, IRC*

1. はじめに

近年の IT の進歩には目覚ましいものがあり，様々な技術が登場，向上していく一方で，価格は急速に低下している．IT の進歩にともない業務の効率化を図り様々な分野に IT が取り入れられるようになった．

多数のビルの状態等を管理するシステムにおいてはクライアントに設置された装置の保守や診断に ICT が取り入れられている．以降ではクライアントに設置された装置をリモート装置と呼ぶ．通常，リモート装置にトラブルが発生した場合は，クライアントからベンダー事業者に連絡が入り，作業員が現場に出向きトラブルの解決を行う．しかし，上記の業務フローでは，作業員が現場に出向いてもトラブルの発生原因がすぐにはわからず，トラブルの解決に現地で多大な時間を消費する場合や，現場に着いても装置が稼働していて，作業は装置を停止させる事の出来る深夜にしか出来ない場合もある．また，装置の電源が入っていないなど，ごく単純な原因で装置が動作しない事をトラブルだと勘違いして連絡を受けた場合は簡単な操作でトラブルを解決できるので現場に出向く時間自体が無駄であり，充分に効率化が図られているとは言えない．上記のような作業はインターネットを通してリモート装置の監視/操作を行うことが可能になれば現地に出向く時間と費用を削減する事が出来，業務の効率化を図る事ができる．

リモート装置の保守/診断の例として NTT ファシリティーズの低価格での監視・保守サービス提供のために開発された監視システムを紹介する．システムの概要図を図 1 に示す．

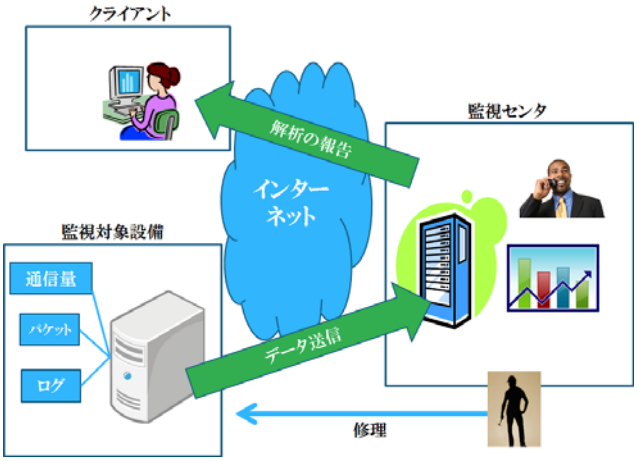

図 1 一般的なリモート端末の保守/診断

このシステムではクライアントに「監視ユニット」と呼ばれる電源装置等を監視するリモート装置を設置し，電源装置にて障害を検知した場合,「センタシステム」と呼ばれる監視センタへ,アラームを随時送信することで，保守センタへの障害対応の駆けつけを手配する．このシステムでは，計測データを定期的に送信しているため現地でトラブルの原因解明に役立てる事が出来，時間の削減が出来る．しかし，監視ユニットのほとんどは，クライアントのローカルネットワーク上に設置されており，監視センタではデータを受信することは出来ても，監視ユニットへアクセスすることが出来ない．また，このシステムは監視センタへのデータの送信に HTTP(Hyper Text Transfer Protocol)を採用している．HTTP は，リクエスト/レスポンス方式でデータを送受信しているので，本システムにおける通信は，リモート装置からのアクセスを契機にしなければ通信を開始できない．また，レスポンスに関しても予め用意した内容を返すことしかできないので，リモート装置からのアクセスを利用した監視センタからの対話的な操作も行うことができない. よって，監視ユニットの操作を要するトラブルに直面した際は，最終的には作業員が現場に出向きトラブルを解決する必要がある．従って，NTT ファシリティーズのシステムのように低価格で提供されているシステムおいては，作業員が現場へ出向くための費用と時間を削減する事が重要な課題となっている．また，クライアントのネットワークに変更を加える手法やシステム用に専用線を引く手法を用いれば，監視センタ等の遠隔地からリモート装置を操作することが出来るので，作業員が直接現場に出向かずにトラブルを解決することが出来るが，初期構築時とランニングの費用が掛かるため導入されにくいという問題がある．そこで，本研究では作業員が現場へ出向く費用と時間の削減を図るために監視センタからもアクセスが可能であり，リモート装置を制御するためのコマンドを監視センタから送信し，リモート装置を操作することのできるシステムを低価格で提供できる手法を提案する. 費用を削減するためにクライアントのネットワークに変更を加えず，公衆回線を利用する実装方法の調査と検討を行い，提案手法の有効な実装方法を示す.

このシステムを構築するためには次の２つの要素が必要である．１つは，リモート装置と監視センタが対話的な通信を行うためのインフラである．もう１つは，監視センタからのコマンドをリモート装置で実行するためのインターフェースである．本稿では，インフラの構築について検討を行った．検討の際に参考にしたシステムは IRC(Internet Relay Chat)プロトコルを利用した BOT Net である. BOT Net とは BOT マスターからの指示を伝えるために BOT に感染した PC で構築されるネットワークのことである．BOT に感染した PC は特定の IRC サーバに接続し，特定のチャンネルに参加する．チャンネルに参加すると，PC は他の PC にユニキャスト，マルチキャストのどちらかでテキストベースのデータを送信することができるため, BOT マスターは同じチャンネルに参加し攻撃の指示をおこなう [1].

IRC プロトコルとは，TCP/IP ネットワーク上でクライアントとクライアントがサーバを経由してテキストデータを交換して会話をするプロトコルである．典型的な構成としては，クライアントの接続点の中心にメッセージの配信, 多重配信などの機能を持つ IRC サーバを設置し，メッセージの交換を行う.

以下本稿では，2 章では，提案手法について述べる．3 章では，実装について述べる．4 章では，実験と評価について述べる．5 章では，評価結果に対する考察を述べる．6 章では，本稿をまとめる.

2. 提案手法

現在リモート装置の保守/診断システムは, 計測データやアラートなどはインターネットを通して通知するが，トラブルが発生した場合は作業員が直接現場に出向き作業を行う手法が主流である．中にはクライアントのネットワークに変更を加えたり，専用線を引いたりして遠隔地からトラブルを解決するシステムも存在するが，初期構築時とランニングの費用が掛かるため導入されにくいという問題がある．そこで，初期構築時とランニングの費用を抑えるために以下に示す 2 点を満たす手法を提案する.

(1) ネットワークに変更を加えず，公衆回線を使用する.
(2) サーバの構築に掛かる費用を抑える.

さらに，同種のリモート装置に対して定期的にメンテナンスを行う場合や同じトラブルに対する処理を 1 対 1 で全て行うのではなく，一斉に処理を行えるように，1 対 1 の通信だけでなく，グループ単位でのマルチキャストを可能にすることで業務の更なる効率化を図ることにする.

この章では，提案システムの概要について述べ，次章で， (1),(2)の条件を満たす 実装方法について述べる.

本システムでは，監視センタにデータ中継サーバとデータの送受信機能を持つコンピュータを設置し，クライアントに本システムでやり取りするデータの処理を行うインターフェースの付いた，テキストデータの送受信機能を持つリモート装置を設置する．以降ではそれぞれの装置を単に監視サーバ, 管理端末, リモート端末と呼ぶ.

それぞれの装置が持つ機能について述べる．まず，監視サーバの持つ機能を以下に示す.

(1)　テキストデータの配送，多重配送機能
(2)　ID と PW によるクライアント認証機能
(3)　SSL による中継データの盗聴・改竄防止機能
(4)　SSL によるサーバ認証機能
(5)　グループ管理機能

(1)の機能はリモート端末から送られる計測データ/アラートを管理端末に，管理端末から送られるコマンドをリモート端末へそれぞれ転送するための機能である．

(2),(3),(4)の機能は本システムが公衆回線を利用する事を前提にしているために必要な機能であり，(2)の機能は管理システムと関係ない端末の接続を防ぐための機能，(3)の機能はクライアント認証情報，計測データ，その他重要な情報を悪用されないようにするため機能，(4)の機能は第三者が設置した偽装サーバに接続する事を防止するための機能である．

(5)の機能は同種の機器，同じ地域にいるクライアントや同一クライアント内に複数のリモート端末を設置しているなど特定の条件下にある複数のリモート端末をグループで管理し，グループ単位でテキストデータを転送する機能である．

次に管理端末，リモート端末の持つ機能について以下に示す．

(1)　テキストデータの送受信機能
(2)　データ送受信用の GUI
(3)　SSL による中継データの盗聴・改竄防止機能
(4)　SSL によるサーバのなりすまし防止機能
(5)　グループ管理機能

(1)の機能はリモート端末から送られる計測データ/アラート，管理端末から送られるコマンドを送受信するための機能である．

(2)の機能は人の手によりデータを送信するための機能と受信したデータを確認するための機能である．

(3),(4)の機能は監視サーバの(3),(4)の機能と同様の機能である．

(5)の機能は管理端末のみが持ち，リモート端末は持たない機能で，グループの追加，作成，削除が可能である．

本システムの構成図を図 2 に示し，1 台のリモート端末を監視する場合の稼動フローを以下に示す．

(1)　リモート端末起動，監視サーバに接続
(2)　通常稼動
(3)　トラブル発生時
(4)　通常稼動

(1)リモート端末は起動すると自動で監視サーバに接続を開始する．この時 SSL を使いサーバの正当性を確認してから暗号化通信を利用して ID と PW を送信し，クライアント認証を行う．クライアント認証後の通信は全て SSL を利用して暗号化して行う．

(2)クライアント認証を済ませた後は，リモート端末としての本来の動作を行い，必要に応じて計測データなど必要な情報を管理端末へ定期的に送信する．また，この時管理端末からのコマンドも受信可能になっているので，コマンドが送られてきた場合はコマンドに従った動作を行う．

(3)トラブルが発生した場合はリモート端末からアラートを送信し，管理端末に異常が発生した事を通知する．アラートを受けた管理端末はコマンドを送信してトラブルの解決を行う．

(4)トラブルが解決したら(2)の通常稼動に戻る．

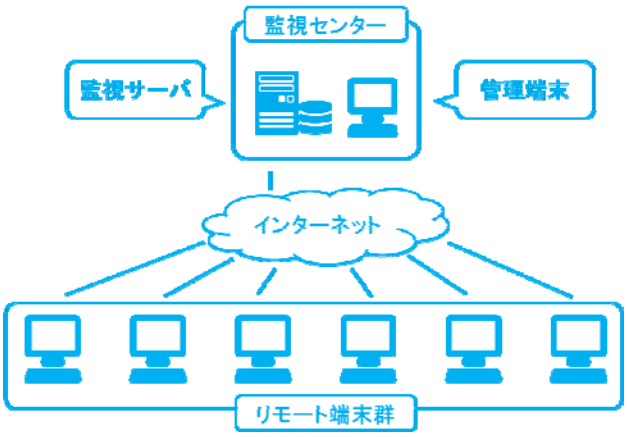

図 2　提案手法の概要図

次にグループを作成して，複数のリモート端末を監視する場合の稼動フローを以下に示す．

監視システムに参加しているクライアントの事業内容と事業所を表 1 に示す．事業所の地域を東北(T)，関東(K)，北海道(H)で表し，事業内容を金融系(F)，環境系(E)，福祉系(W)で表す．各企業の事業所には事業内容ごとのリモート端末がそれぞれ 1 台ずつ設置されているものとする．

表 1　各企業の事業所と事業内容

企業名	A	B	C	D
事業所	T	K	T	T
	K	H	K	H
事業内容	F	F	E	F
	E	E	W	W
	W	―	―	―

表 1 のクライアントをグループで管理する場合のグループ例を企業別，地域別，事業内容別に表 2，表 3，表 4 に示す．

表 2　企業別のグループ例

グループ	A	B	C	D
企業名, 事業所, 事業内容	A,T,F	B,KF	C,T,E	D,T,F
	A,T,E	B,K,E	C,T,W	D,T,W
	A,T,W	B,H,F	C,K,E	D,H,F
	A,K,F	B,H,E	C,K,W	D,H,W
	A,K,E	―	―	―
	A,K,W	―	―	―

表 3　地域別のグループ例

Table 1　Examples of groups of each region

グループ	H	K	T
企業名, 事業所, 事業内容	B,H,F	A,K,F	A,T,F
	B,H,E	A,K,E	A,T,E
	D,H,F	A,K,W	A,T,W
	D,H,W	B,K,F	C,T,E
	―	B,KE	C,T,W
	―	C,K,E	D,T,F
	―	C,KW	D,T,W

表 4　事業内容別のグループ

グループ	F	E	W
企業名, 事業所, 事業内容	A,T,F	A,T,E	A,T,W
	A,K,F	A,K,E	A,K,W
	B,K,F	B,K,E	C,T,W
	B,H,F	B,H,E	C,K,W
	D,T,F	C,T,E	D,T,W
	D,H,F	C,K,E	D,H,W
	―	―	―

(1)　全てのリモート端末起動，監視サーバに接続
(2)　管理端末がグループを作成
(3)　通常稼動
(4)　金融系のリモート端末でトラブル発生
(5)　通常起動

(1)1 台のリモート端末を管理する場合と同様に SSL を使って全てのリモート端末の認証を行う．

(2)管理端末から表 2, 表 3, 表 4, に示すように企業別，事業所別，事業内容別のグループを作成し，ユーザの追加を行う．1 つのリモート端末は複数のグループに重複して参加させることが出来る．グループの作成，追加，削除は管理端末からのみ行える．また，管理端末からの意思のみでリモート端末の同意を必要としない．また，グループの作成，追加，削除は動的に行え，(3)の通常起動に状態が移った後にも行える．

(3)1 台のリモート端末を管理する場合と同様に計測データなど必要なデータを管理端末にのみ送信する．この時同じグループに所属しているリモート端末には計測データ等は中継されない．

(4)金融系のリモート端末に 1 台にトラブルが発生した場合，リモート端末からアラートがリモート端末に送信される．アラートを受けた管理端末はアラートを送信してきたリモート端末だけでなく同じグループに対して診断/操作を行う. 全ての操作が完了したら(2)の通常稼動に戻る．

3.　実装

この章では，提案システムの性能要件と機能要件をまとめ，機能要件で挙げた機能の実装について述べる．

3.1　性能要件

提案システムが満たすべき機能要件を以下に示す．

(1)　ネットワークに変更を加えず, 公衆回線を利用する．
(2)　サーバの構築費を抑える
(3)　同時接続数が 500~1000 台
(4)　レスポンスタイムが 10 秒以内

3.2　機能要件

提案システムが満たすべき機能要件を以下に示す．

(1)　リモート装置と管理センタの双方向通信
(2)　ID と PW によるクライアント認証
(3)　SSL による保護
(4)　グループ機能

3.3 実装環境

提案システムを実装した環境を表 5 に示す.

表 5　提案システムの実装環境

OS	サーバ	データベース	通信プロトコル
Windows7 32bit	Node.js 0.11.11	MongoDB 2.6.4	WebSocket [2]

3.4 実装機能

(1)　リモート装置と監視センタの双方向通信

Node.js でサポートされている WebSocket のための通信モジュールである socket.io を利用して実装を行った．ConnectionUserList クラスを作成して相手を指定してメッセージを送信できるように実装した.

(2)　ID と PW を利用したクライアント認証

WebSocket のコネクションを確立する前に HTTPS で ID と PW による認証を行った後 Cookie に認証情報を保持させ WebSocket でも認証情報を取得できるようにする．HTTPS での ID と PW による認証は HTTPS でクライアントから ID と PW を POST し，MongoDB に登録されている ID と PW 照合して認証を行うように実装した.

(3)　SSL による保護

Node.js では HTTPS による接続がサポートされている．HTTPS でコネクションを確立するためには秘密鍵と証明書付きの公開鍵が必要なので，OpenSSL で秘密鍵と公開鍵を作成して自己署名の証明書を作成した．証明書付きの公開鍵を使用することで，サーバのなりすましを防ぎ，HTTPS を利用して通信することでクライアント認証情報や計測データなどの送受信データの盗聴改竄を防止した.

(4)　グループ機能

socket.io にも room と呼ばれるグループ機能があるが，IRC プロトコルと同様に動的に作成されたグループに参加できない問題と，システムに参加している計測データなどが漏洩してしまう問題がある．そこで，Group クラスを作成してグループ機能を実装した.

4.　評価

この章では，3 章の実装で挙げた性能要件と機能要件を実装したシステムが満たしているか検証を行った.

4.1 性能評価

ネットワークに変更を加えず，公衆回線を利用する点についてはポート 443 番を利用する WebSocket を使用して実装することで満たすことができた．その他の性能要件について，サーバの構築費を抑える，同時接続数 500～1000 台以上，レスポンスタイム 10 秒以内という点について要件を満たしているか評価するために実験を行った．行った実験内容を以下に示す.

(1)　100~1000 のクライアントをサーバに接続した時にサーバで使用される CPU 使用率を計測した.

(2)　100~1000 の全てのクライアントに対して同時に通信を行った時にサーバで使用される CPU を計測した.

(3)　100~1000 の全てのクライアントに対して同時に通信を行った時に掛かる時間を計測した.

実験内容(1)に示す実験結果を図 3 に，実験内容(2)に示す実験結果を図 4 に, 実験内容(3)に示す実験結果を図 5 に示す.

CPU の使用率については，リモート装置の同時接続数が 100~1000 台では，待機時で最大 3%，データ送信時で最大 21%であった．従って，性能要件のサーバの構築費を抑える点と同時接続台数が 500~1000 という点は満たしていることが確認できた．レスポンスタイムについては，リモート装置の接続数が 100~1000 台では 16ms と一定の値で安定しており，性能要件のレスポンスタイムが 10 秒以内という点も満たしていることが確認できた.

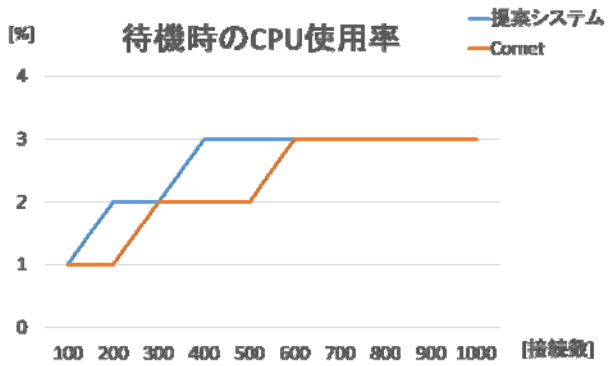

図 3　待機時の CPU 使用率

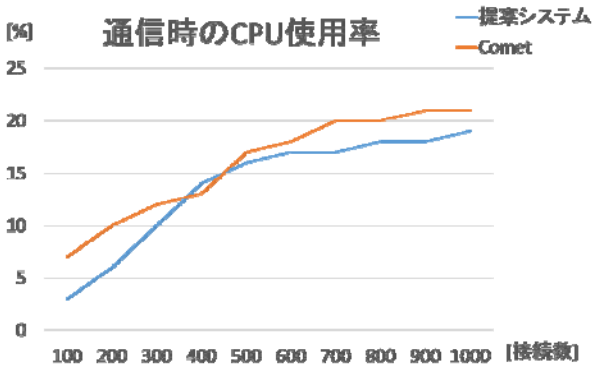

図 4　全てのクライアントにデータを送信したときの CPU 使用率

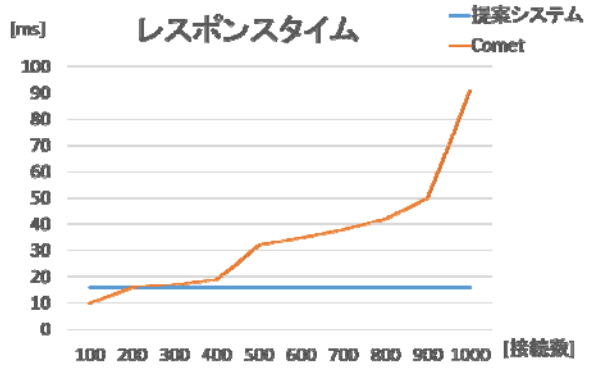

図 5　全てのクライアントにデータを送信したときのレスポンスタイム

4.2 機能評価

3 章の実装で挙げた 4 つの機能要件について提案システムの動作フローに沿って検証を行った．クライアントにはリモート装置のとして WebSocket に対応しているブラウザを用いて実験を行った．行った実験内容を以下に示す．

(1)　リモート装置から監視センタに接続を行い SSL による保護ができているか検証行った．

(2)　リモート装置から ID と PW にようクライアント認証済みのクライアント以外がシステムに参加できるか検証を行った．

(3)　リモート装置と監視センタの管理端末で相互のメッセージを送受信し，リモート装置と監視センタの双方向通信が可能か検証を行った．

(4)　管理端末と ID が honma,ootuka,kaji のリモート装置を監視センタに接続して honma,ootuka をグループにし，管理端末からそのグループにメッセージを送信してグループ単位でメッセージの送信が可能か検証を行った．

実験内容(1),(2),(3),(4)の実験を行った結果, 機能要件に挙げた 4 つの機能について問題なく実装できていることが確認できた．

5. 考察

リモート装置の接続数が 100~1000 台ではサーバの CPU 使用率が最高でも 21%であったため充分にサーバの初期構築費用を抑えることが出来るので，結果としてサービスを安価で提供できると考えられる．また，クライアントのネットワーク変更を加えずシステム用に専用線を引かず，多くの会社で利用されることが許可されている 443 ポートを使用してデータ通信を行うことが可能なので安価にサービスを提供できると考えられる．

リモート装置の接続数が 100~1000 台では全ての接続されたリモート装置にデータを送信した場合でもレスポンスタイムが安定して 16ms であったことから，1 対 1 の通信はもちろんグループを作成し，複数のリモート装置に場合でも全てのリモート装置にデータを送信する場合でも 16ms を上回ることはないと考えられるため，レスポンスタイムに関する性能要件は充分に満たしていると考えられる．

実験結果からわかる通りリモート装置の接続数が 100~500 台に増えた時の待機状態と接続された全てのリモート装置にデータを送信した時の CPU の使用率の上昇に比べリモート装置の接続数が 500~100 台に増えた時の待機状態と接続されて全てのリモート装置にデータを送信した時の CPU の使用率の上昇率が緩やかになっていることから

コスト面と同時接続耐性については接続台数をまだ増やすことができると考えられる．

6. 結論

本研究ではリモート装置の保守/診断の業務を遠隔地からコマンドでリモート装置の操作を可能にするという点と 1 対 1 の操作だけではなくグループ単位によるマルチキャストでコマンドを送信できるようにするという点で効率化を図る手法を提案した．

提案システムを構築するためには次の２つの要素が必要である．１つは，リモート装置と監視センタが対話的な通信を行うためのインフラである．もう１つは，監視センタからのコマンドをリモート装置で実行するためのインターフェースである．本研究では，インフラの構築について実装を行った．提案手法が要求する性能要件と機能要件を実装したシステムが満たしているか実験を行って検証した．検証した結果全ての要件を満たしていることが確認できた．しかし，実験にはリモート装置の代わりに WebSocket をサポートしているブラウザを使用し

て行ったので，今後は監視センタからのコマンドをリモート装置で実行するためのインターフェースを実装し，短期間でも良いので実際に複数のリモート装置の管理を行い性能評価と機能評価を行っていく必要がある. また，グループ単位による管理が可能なので，どのようなグループを作成して管理したらリモート装置の保守/診断の効率を上げられるか検討を行う必要がある.

7. 謝辞

本研究を進める上で協力してくださった金井研究室の皆様に，心からお礼申し上げます．御尽力頂きありがとうございました.

本稿をまとめるにあたり協力してくださった NTT ファシリティーズの川上公一郎様,中尾貢様に感謝申し上げます．ご尽力いただき本当にありがとうございました.

8. 参照文献

[1] 荒谷　光, 金井　敦,斉藤　典明,本間, 将紘, “IRC プロトコルを利用した攻撃者と感染端末の探索手法, ” コンピュータセキュリティシンポジウム 2013 論文集, 2013.

[2] I. Fette, “Request for Comments 6455: The WebSocket Protocol, ” 2011. [オンライン]. Available: https://tools.ietf.org/html/rfc6455.