認証済み Cookie 情報の端末間での連携技術の開発と評価

梅澤 克之 †　　加藤 崇利 †　　田代 卓 †

†(株) 日立製作所 システム開発研究所
244-0817 神奈川県横浜市戸塚区吉田町 292 番地

{katsuyuki.umezawa.ue, takatoshi.kato.bb, takashi.tashiro.my}@hitachi.com

あらまし 近年様々な端末を使って様々なサービスが受けられるようになってきた．たとえば，外出時には携帯電話端末で社内業務を行い，帰社後は，大きな画面の PC で引き続き業務を継続できると便利である．しかし，現状では帰社後に PC で再び認証を受ける必要があった．本論文では，社外から社内サーバを利用する業務形態において，外出時には携帯電話端末を利用し，帰社後には PC を利用する場合に，ユーザが端末を切り替える際の認証方式を提案する．具体的には，認証済みの情報としての Cookie 情報を引き継ぐことによって，旧端末から新端末に切り替えたときに，サーバ側の認証を簡略化する方法を提案し，提案に基づいたシステムを開発し性能を評価する．

Development and Evaluation of Coordination Technology using Cookie as Authenticated Information between Terminals

Katsuyuki Umezawa†　　Takatoshi Kato†　　Takashi Tashiro†

†Hitachi, Ltd. Systems Development Laboratory
292, Yoshida-cho, Totsuka-ku, Yokohama-shi, Kanagawa, 244-0817 Japan

{katsuyuki.umezawa.ue, takatoshi.kato.bb, takashi.tashiro.my}@hitachi.com

Abstract Recently, a lot of users have come to use the cellular phone terminal. Moreover, a variety of electronic equipment have come to be connected continuously with a high-speed broadband line. In such a situation, it is convenient that service is received with the cellular phone terminal when we go out, and service is received continuously by a PC of a large screen after we come office. It is more convenient if the authentication of PC becomes unnecessary after the cellular phone is authenticated. In this paper, we propose the authentication method when the user switches the terminal in the environment that the user receives service with a terminal from server. Concretely, we develop and evaluate the system to simplify the authentication of the server side when switching from an old terminal to a new terminal by succeeding the cookie as authenticated information.

1　はじめに

近年，携帯電話端末装置が普及し，多くのユーザが利用するようになってきた．また，様々な電子機器が高速なブロードバンド回線に常時接続され，様々なサービスを受けられるようになってきた．このような状況において，例えば外出時に携帯電話端末で社内業務を行い，帰社後は，大画面の PC で引き続き業務を継続できると便

利である．携帯電話端末からのアクセスが認証された後では，PC からのアクセス時に認証が不要になるなら，より便利である．

本論文では，今まで使っていた端末（スマートフォン端末）とこれから使う端末（PC 端末）の間で認証済み情報としての Cookie 情報を連携することにより，新端末（PC 端末）とサーバの間の再認証処理を不要にした認証連携システムを提案する．

まず 2 章で，従来技術として Cookie 情報を用いた認証方式，および NFC(Near Field Communication) による接続ハンドオーバー技術について記述する．3 章で提案方式を記述する．具体的にはスマートフォン端末と PC 端末間での Cookie 情報の連携フローとそれを実現するための機能について記述する．4 章で提案方式に従って実装したシステムの実行時間の評価を行う．最後に 5 章でまとめと今後の課題を示す．

2　従来技術

本節では，従来技術として Cookie 情報による認証方式，NFC(Near Field Communication) による端末間のハンドオーバー技術，および筆者らが提案しているスマートフォンを用いたリモートアクセス技術について記述する．

2.1　Cookie 情報による認証方式

Cookie は HTTP プロトコルを用いた Web ブラウザ間での状態を管理するプロトコルとして使用されている．主な目的は，情報の再利用である．例えば Web サイトの訪問履歴やログイン情報など Web ブラウザによって保存され，再度同じ Web サイトを訪問した際に保存された Cookie 情報を Web サイトに送信することによって，訪問履歴の更新やログイン処理を省略することが可能となる．

Cookie については，RFC2109[1]，RFC2965[2] で規定されている．RFC2109 には「Cookie はクライアントからサーバへリクエストが送信された際に，Cookie を HTTP レスポンスヘッダの Set-Cookie ヘッダに設定してクライアントに送信できる」と記述されている．また，クライアントが Cookie を保持している場合，HTTP リクエストに保持している Cookie を設定してサーバに送信することができる．

Cookie を用いた認証処理の基本フローを図 1 に，その説明を表 1 にそれぞれ示す．

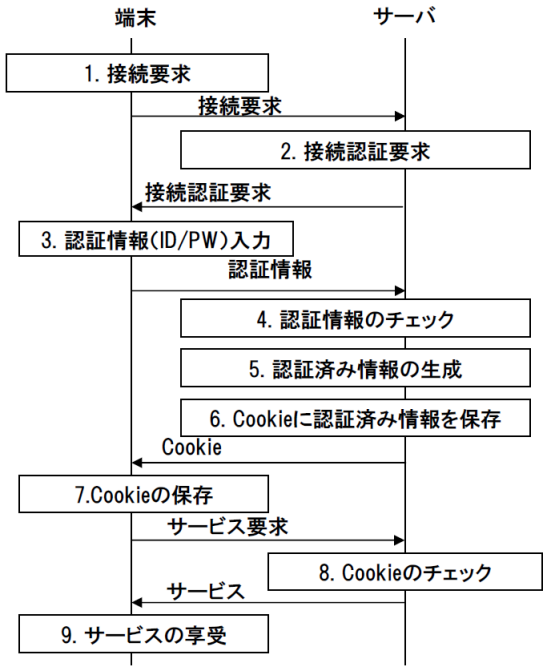

図 1: Cookie を用いた認証処理の基本フロー

2.2　NFC による接続ハンドオーバー技術

NFC による接続ハンドオーバーとは，NFC Forum で規定している，端末－端末間での接続技術であり，端末と端末の対応関係の確立（ペアリング）だけを NFC(ISO 14443 Type A/Type B 等) で行い，以降のデータ通信は，さらに高速な Bluetooth や WiFi を行う技術である [3]. 図 2 に NFC による接続ハンドオーバー技術の概要を示す．

2.3　スマートフォンを用いたリモートアクセス技術

筆者らは，スマートフォン端末をセキュリティデバイスと見なして PC 端末と連携させてリモートアクセスを行うシステムの提案を行って

表 1: 図 1 の説明

No.	説明
1	端末からサーバへ接続要求を行う.
2	接続要求を受けたサーバは，端末に対して接続認証要求を行う.
3	接続認証要求を受けた端末で，認証情報 (ID，パスワードなど) の入力を行い，入力した認証情報をサーバへ送信する.
4	認証情報を受けたサーバは登録されている認証情報と端末送信された認証情報とが一致するかチェックする.
5	認証情報が一致していれば，サーバは認証済み情報の生成を行う.
6	生成した認証情報を Cookie に保存し，端末へ送信する.
7	端末は Cookie を保存する．再接続時には，端末は，サーバへサービス要求，Cookie の再送信を行う.
8	サービス要求と Cookie を受けたサーバは，Cookie に保存されている認証済み情報とサーバが生成した認証済み情報とが一致するかチェックを行う．その後サービスを提供する.
9	端末は，サービスを受ける.

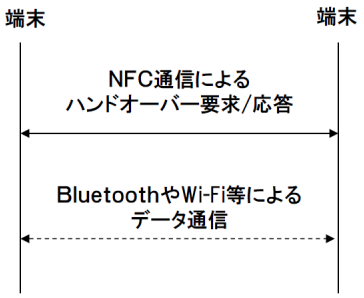

図 2: NFC による接続ハンドオーバーの概要

きた [4][5][6]．しかし，これらの提案ではスマートフォン端末と PC 端末は個人の持ち物という前提でそれらの端末の組み合わせは固定的であった．例えば共有 PC 端末を利用する場合などは動的な端末の組み合わせが必要とされていた.

3 提案方式

本節では，スマートフォン端末と PC 端末間で認証済み情報を連携することにより，PC 端末とサーバの認証処理を簡略化した認証連携システムを提案する.

3.1 提案方式の概要

提案方式の全体概要を図 3 に示す．まず，ユーザが，駅のホームなど PC を広げられない時にはスマートフォンを単体で用いて社内システムにログインし，認証を受けた後に業務を行う．その後，PC を広げられる状況になった場合には，スマートフォンを PC にかざすだけで，事前にスマートフォン単体で受けている認証済み情報を PC 側に転送することで，再度認証処理を行うことなく業務を再開できる.

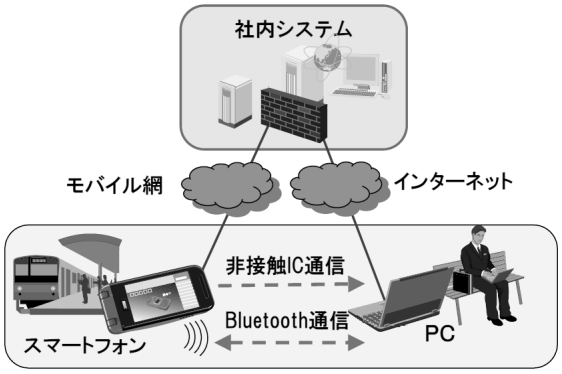

図 3: 提案方式の全体概要

3.2 提案方式の処理フロー

3.2.1 スマートフォン単体での処理

まず，スマートフォン単体を用いて社内システム（サーバ）にログインする際の処理を説明する．図 4 にフローを示す．まず，スマートフォン側のブラウザを用いて社内ウエブサイトにアクセスする[1]．次に初期の認証処理を行うために認証情報として ID，パスワードを送信する[2]．サーバ側で認証処理を行った後，認証済み情報として Cookie 情報が発行される．スマートフォン側のブラウザは，Cookie 情報を保存する.

[1] 今回の社内システムは Web ベースのシステムを想定している.

[2] 今回認証方法は ID/パスワード方式を採用しているが，PKI（公開鍵暗号）方式に基づく認証方式でも良い.

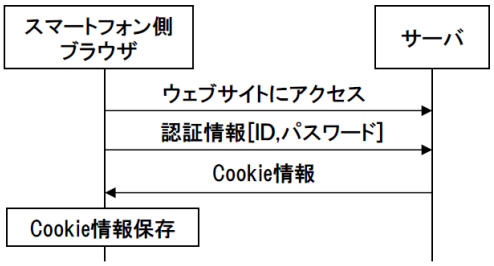

図 4: スマートフォン単体での処理

3.2.2 スマートフォンと PC の連携処理

次に，スマートフォンから PC に業務を引き継ぐ際の Cookie 情報の連携処理を説明する．図 5 にフローを示す．前提として，スマートフォンは，図 4 に示したような初期の認証処理は実行済みであり，スマートフォン内に Cookie 情報が保存されていると仮定する．

（処理 1）まずスマートフォンを PC にかざす．PC 側のアプリケーションは，非接触 IC 通信を用いてスマートフォン内の非接触 IC チップの ID を読み込む．（処理 2）次に PC 側のアプリケーションは，前述の非接触 IC チップの ID の変換値を Bluetooth のペアリングのための PIN コードとして用いて，スマートフォン側アプリケーションとペアリング処理を実行する．この際スマートフォン側のアプリケーションも同様に非接触 IC チップの ID の変換値を Bluetooth のペアリングのための PIN コードとして用いる．（処理 3）ペアリングが完了すると，PC 側アプリケーションは Bluetooth 通信で Cookie を要求し，スマートフォン側アプリケーションから提示された Cookie 情報を，PC 側ブラウザの DB に格納する．（処理 4）その後，PC 側ブラウザを起動し，前述の Cookie 情報とともに社内システム（サーバ）にアクセスする．サーバは，Cookie 情報をチェックした後，サイト情報を送信し，PC 側ブラウザはサイトを表示する．

3.3 提案システムの機能

3.3.1 PC 側アプリケーションの機能

PC 側アプリケーションは，非接触 IC リーダからの読み込みシグナルをトリガーとして動作

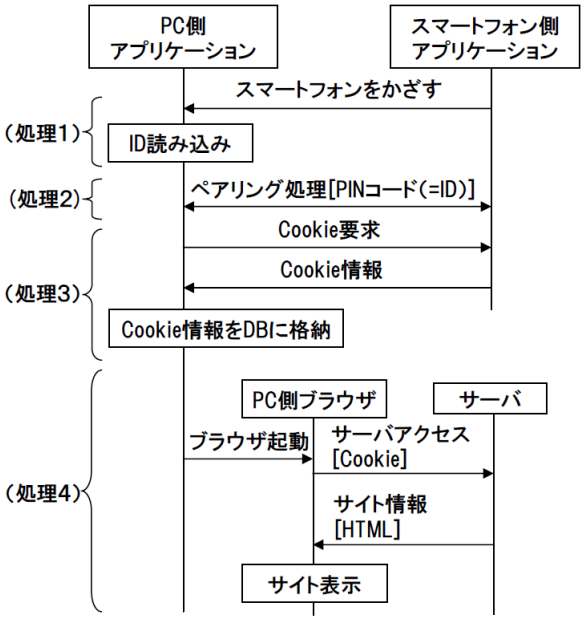

図 5: スマートフォンと PC の連携処理

し，スマートフォンとの Bluetooth ペアリング処理，スマートフォン側アプリケーションからの Cookie 情報と URL の受け取り，およびブラウザ起動処理を行う．以下に今回実装した提案システムの各機能を記述する．

(1) 非接触 IC チップの ID の読み込み機能 本機能は非接触 IC リーダにスマートフォンがかざされたときに動作し，非接触 IC チップの ID を取得する機能を提供する．

(2)Bluetooth ペアリング機能 本機能は，PC とスマートフォン 2 台の Bluetooth 機器のペアリングを自動的に行う機能を提供する．ペアリングにおいては非接触 IC チップの ID の読み込み機能で取得した ID から生成した 16 バイトの PIN コードを利用する[3]．

(3)Cookie 情報インポート機能 本機能は，スマートフォン側アプリケーションから送信された Cookie 情報を取得し，PC 側のブラウザにインポートする機能を提供する．

[3]接続相手の決定は，理想的には上述の「非接触 IC チップの ID の読み込み機能」を使って ID とともに Bluetooth アドレスを取得することが望ましいが，今回の実装では行っていない．

今回の実装では Firefox の Cookie データベースを直接操作することで，Firefox に対して Cookie 情報のインポートを行う．

(4) ブラウザ起動機能 本機能は，Firefox を別プロセスで起動してスマートフォン側アプリケーションから受け取った URL を表示する機能を提供する．

3.3.2 スマートフォン側アプリケーションの機能

スマートフォン側アプリケーションはバックグラウンドで動作し，起動するとすぐに PC 側アプリケーションからの接続要求待ち状態に入る．以下に今回実装した提案システムの各機能を記述する．

(1) 非接触 IC チップの ID 取得機能 自端末に組み込まれている非接触 IC チップの ID を取得する機能を提供する[4]．

(2)Bluetooth ペアリング機能 本機能は，PC とスマートフォン 2 台の Bluetooth 機器のペアリングを自動的に行う機能を提供する．ペアリングにおいては非接触 IC チップの ID 取得機能で取得した ID から生成した 16 バイトの PIN コードを利用する[5]．

(3)Cookie 情報エクスポート機能 本機能は，Internet Explorer Mobile が管理している Cookie 情報を取得し，Cookie 送信用のデータフォーマットに変換して PC 側アプリケーションに送信する機能を提供する．

(4)URL の取得と送信機能 Internet Explorer Mobile で表示している URL を取得し，PC 側に送信する機能を提供する[6]．

[4]本機能は自端末に組み込まれている非接触 IC チップから ID を取得することが望ましいが，今回の実装では，あらかじめレジストリ設定された ID を取得している．

[5]Bluetooth のペアリングでは接続される側の端末も相手のアドレスを指定する必要がある．今回の実装では，スマートフォン側で PC 側からのソケット接続を待ち，接続後にソケット接続情報から PC 側の Bluetooth アドレスを取得した後に，ペアリング処理を実行している．

[6]今回の実装では，Internet Explorer Mobile からの URL 取得が実現できなかったためにレジストリに URL をハードコーディングしている．

4 実行時間の評価

本節では，提案方式に従って開発したシステムを用いて実行時間の評価を行う．

4.1 測定条件

性能測定対象の端末のスペックは，表 2，3 に示した通りである．また，端末間の非接触 IC 通信には FeliCa を用い，ハンドオーバー後の Bluetooth に関しては Bluetooth Ver.2.1 対応のアダプタを用いた．さらに，PC とサーバ間のネットワークは 8Mbps の ADSL 通信網を用いた．

表 2: 測定に用いた PC のスペック

OS	Windows XP SP2
CPU	Intel Core2 Duo T8100(2.10GHz)
メモリ	3GB
ブラウザ	Firefox 3.0.7

表 3: 測定に用いたスマートフォンのスペック

OS	Windows Mobile 6 Standard Edition
CPU	ARM1136 OMAP2430(330MHz)
メモリ	256MB(ROM)/128MB(RAM)
ブラウザ	Internet Explorer Mobile

4.2 測定項目

測定対象の処理は，下記のとおりである．なおそれぞれの処理は，図 5 における同名の処理に対応する．

- スマートフォンを PC にかざしてから ID を読み込むまでの時間（処理 1）
- ペアリング要求を出してから，ペアリングが終了するまでの時間（処理 2）
- Cookie 要求を送信してから Cookie を DB に格納するまでの時間（処理 3）
- ブラウザを起動してサイトを表示するまでの時間（処理 4）

4.3 測定結果

測定結果を表 4 に示す．各測定項目において 12 回測定し，全体の時間が最小と最大のデータを除外した 10 回分の測定値から平均値を算出した．

表 4: 測定結果

測定項目	測定結果 (秒)
処理 1	0.31
処理 2	0.76
処理 3	1.49
処理 4	1.64
全体	4.19

表4からわかるように，ユーザがスマートフォン端末を PC 端末にかざしてから約 4 秒で PC 端末を用いてサービスを受けられることが確認できた．

5 まとめと今後の課題

本論文では，ユーザが今まで使っていたスマートフォン端末とこれから使う PC 端末の間で認証済み情報としての Cookie 情報を連携することにより，連携後の PC 端末とサーバの間の再認証処理を不要にした認証連携システムを提案した．さらに提案に基づいて実装したシステムの実行時間を計測し，スマートフォン端末を PC 端末にかざしてから約 4 秒で新端末からサービスを受けられることを示した．

今後は，今回の実装では省略した端末同士の正当性の検証技術の提案，および Cookie 情報以外の認証済み情報の連携技術の提案を行う予定である．

謝辞 本研究は, 独立行政法人情報通信研究機構 (NICT) の委託研究「端末プラットフォーム技術に関する研究開発」の成果の一部である．

参考文献

[1] IETF RFC2109 HTTP State Management Mechanism

[2] IETF RFC2965 HTTP State Management Mechanism

[3] Connection Handover Technical Specification, NFC Forum, Nov. 2008

[4] 梅澤克之, 洲崎誠一, “スマートフォンを用いたリモート接続システムの開発,” 第 31 回情報理論とその応用シンポジウム予稿集, pp.971–974, Oct. 2008.

[5] 梅澤克之, 加藤崇利, 手塚悟, “携帯端末を用いた FMC 認証方式の開発,” 電子情報通信学会 技術研究報告 (ISEC2009-36, SITE2009-28, ICSS2009-50), pp.203–208, Jul. 2009.

[6] 梅澤 克之, 加藤 崇利, 手塚 悟, “スマートフォンを用いたリモート接続システムの開発と評価,” 第 8 回情報科学技術フォーラム (FIT2009) 予稿集 第 4 分冊, pp.67-73, Sep. 2009