IMAPエージェントによるサーバの設定変更を要しない柔軟なメール転送システム

田代 寛三[1]　山井 成良[1]　北川 直哉[1]

概要：電子メールの転送処理はMTA（Mail Transfer Agent）の設定やMDA（Mail Delivery Agent）の設置による実現が一般的である．しかし，そのような直接的なメールサーバの設定変更が認められないフリーメールサービス等においては，サービス運営者がWebサービス等を介して提供するGUIの範囲外の高度な転送処理を行うことができない．このように現行の転送システムでは，転送処理の柔軟性がメールサーバに対するユーザの権限に左右される．そこで本稿では，メールサーバに対するユーザの権限に依存せずに柔軟な転送処理を実行でき，かつメールサーバの設定変更を必要としない新たなメール転送システムを，IMAP（Internet Message Access Protocol）エージェントによって実現する．

Flexible Email Forwarding System Using IMAP Agent without Changing Server Configuration

KANZO TASHIRO[1]　NARIYOSHI YAMAI[1]　NAOYA KITAGAWA[1]

1. はじめに

電子メールは個人や企業等，利用者の種別を問わず広く利用されており，一利用者が複数のメールアカウントを取得，管理していることも少なくない．そのような利用者に向けて，受信したメールを別のメールアドレスに自動転送する機能を提供するサービスやソフトウェアも既に存在し，転送を実行する機構もメールサーバに備わっていることが一般的である．

例えば，Sendmail[1]やPostfix[2]といった代表的なMTA（Mail Transfer Agent）では，ユーザのホームディレクトリに転送先のメールアドレスを列挙したファイルを設定することで自動転送を実現できる．このファイルには外部ツールやシェルコマンドを転送時の処理に組み込むことができ，柔軟な転送処理を設定することが可能である．また，メールサーバにmaildrop[3]等のMDA（Mail Delivery Agent）を設置することでも同様の処理を実現できる．ただし，これらの高度な転送設定は，利用者がメールサーバに対する直接的な書き込みの権限を有している場合にだけ可能である．一般的に，広く普及しているフリーメールサービス等においてはそのような権限は利用者に与えられておらず，Webサービス等を使用してメールサーバの転送設定を行う．この場合，利用者は用意されたGUIを介して転送ルールを設定することになるため，容易に設定を行える一方で，メールサービス運営者が実装し提供する範囲外の高度な転送処理は行えない．

そこで本論文ではIMAP（Internet Message Access Protocol）[4]に対応したメールサーバ（以下，IMAPサーバ）を対象に，利用者の権限にかかわらず柔軟な転送処理を実行できるメール転送システムを，メールサーバに対する設定変更を必要としない形で提案する．本システムにおけるメールの自動転送処理はIMAPエージェント[5]が担う．IMAPエージェントはメールサーバに常時接続することで，メールが接続先のサーバに到着すると直ちにその内容を取得し，その内容に応じた処理を可能にするシステムである．このIMAPエージェントが転送元のメールサーバとの間に常時接続を築き，新たなメールを検知すると事前に登録した処理を実行したのち，転送先のメールサーバに該当メールを直接保存することで転送を実現する．このメール検知から転送までに行える処理の実装に特段の制限はなく柔軟な転送処理が可能となる．さらに，転送を担うシス

[1] 東京農工大学

© 2019 Information Processing Society of Japan

テムと転送に関わる情報を管理するシステムを分けることで，より実用的なメール転送システムの構築を行う．

また，本システムの柔軟な転送処理の一例として，送信ドメイン認証を実装しシステムの検証を行う．具体的には，SPF（Sender Policy Framework）[6]，DKIM（DomainsKey Identified Mail）[7] および DMARC（Domain-based Message Authentication, Reporting, and Conformance）[8] の機能を実装する．これらの処理によって各種送信ドメイン認証に対応していない転送先に対してセキュアな転送手段を提供することに加え，転送時に起こりうる認証上の問題を一部解決することができる．

本稿では，2 章で関連技術として従来の転送手法と IMAP エージェントについて説明し，3 章では提案するシステムの概要と詳細な設計について述べる．4 章では試作したシステムの動作検証を行い，5 章でシステムの実用化に向けた議論を行う．最後に 6 章で本稿をまとめる．

2. 関連技術

2.1 従来の転送手法

2.1.1 MTA による転送

Sendmail や Postfix 等の代表的な MTA では，ユーザが自分のホームディレクトリに転送ルールを記述したファイル（~/.forward）を用意することで，自動転送を実現することができる．ファイルには転送先メールアドレスを列挙できるほか，パイプラインによって受信メール内容を外部のプログラムに引き渡すことで，様々な転送処理を実行できる．

2.1.2 MDA による転送

代表的な MDA の一つである maildrop では，独自のスクリプトファイルを指定のパスに設置することで，メールのヘッダや本文に対する条件分岐を用いた転送ルールを設定することができる．また，MTA の場合と同様にパイプラインにより外部プログラムにメールデータを引き渡して実行することも可能である．

2.1.3 従来手法の欠点

MTA または MDA どちらの転送手法においても，ユーザが自由に設定を行うにはサーバに対する直接的な編集権限が必要になる．しかし，外部プログラムを実行できてしまうセキュリティ上の問題や，適切な設定にスクリプトファイルの構文を理解する必要があるという点等から，多数のユーザが利用するようなメールサービス，例えば Gmail や Outlook といったフリーメールサービスにおいて，ユーザにそのような権限は与えられていない．そのような場合は一般的に，専用のメールクライアントや Web サービス等を介した転送ルールの設定手段を提供している．転送ルールの設定は GUI 操作によって行われることが多く，設定できる内容はメールサービス運営者が実装し提供する範囲に限られる．したがって，現存する方式で柔軟な自動転送機能を実現できる状況は，ユーザがメールサーバに対する直接的な編集権限を有している場合に限られ，転送設定の柔軟性はメールサーバに対するユーザの立場に大きく左右されることになる．

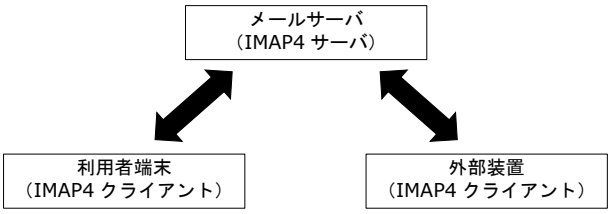

図 1 独立型の IMAP エージェント

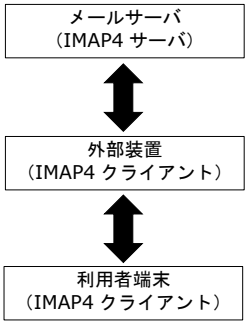

図 2 プロクシ型の IMAP エージェント

2.2 IMAP エージェント

IMAP エージェントは IMAP サーバとの間に常時 IMAP セッションを保つことで，新たなメールの到着を検知し，そのメールに対して任意の処理を実行できる機構である．提案されている IMAP エージェントの構成には独立型とプロクシ型の二種類が存在し，それぞれの構成を**図 1** および**図 2** に示す．

どちらの構成においても，IMAP エージェントは IMAP サーバおよび利用者端末とは別に設けた外部装置で IMAP クライアントとして動作する．新たなメールの検知は IDLE コマンド [9] を使用したステータス更新通知の受信または NOOP コマンドによる IMAP サーバのステータス情報のポーリングによって実現する．

3. IMAP エージェントを用いたメール転送システム

3.1 提案システムの概要と動作

本提案は，従来の転送手法におけるユーザ権限への依存を解消し，かつ転送元および転送先のメールサーバの設定を変更せずに，メール転送時に柔軟な処理を実行可能とするシステムの構築を目標とする．また，実用化を意識したシステム構築を目指し，多数のユーザが本システムの転送機能を利用できるように設計を行う．

この目標を達成するうえで，メールサーバに対して設定変更を強制しないためには外部装置が転送機能を担うこと

© 2019 Information Processing Society of Japan

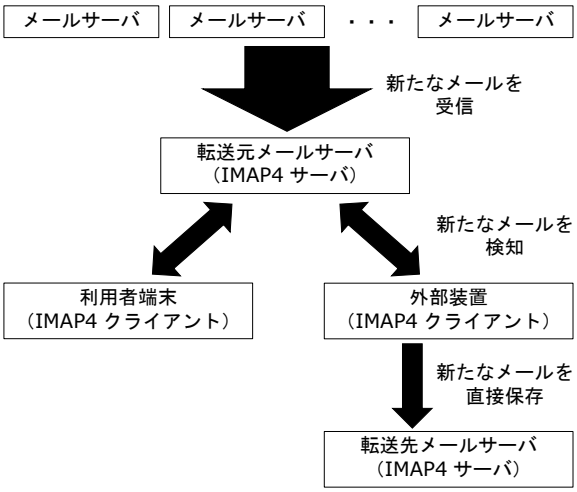

図 3 IMAP エージェントによるメール転送

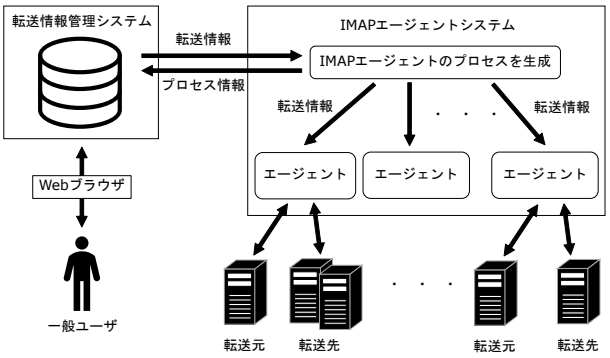

図 4 メール転送システムの概観

が求められる．そこで，IMAP サーバに対する常時接続によってメールの到着検知を行える IMAP エージェントを起用する．なお，転送を担う本システムでは，図 2 のプロクシ型のように，新たなメールに対して利用者端末よりも必ず先にアクセスする構造をとる必要性は低いと考え，図 1 の独立型を採用する．また，独立型の採用により，ユーザが転送元のメールサーバと行う通信を中継する必要がなくなることから，外部装置が占有するリソースを削減することができ，多数ユーザへの転送機能の提供を目標とした本システムにも適していると考えられる．

独立型の IMAP エージェントによる転送は以下の手順で実現する．なお，転送処理時の構成は**図 3** の通りである．

(1) 転送元 IMAP サーバに接続を行い，IMAP セッションを維持する．
(2) IMAP サーバのステータスが更新されるまで待機する．
(3) 新たなメールが存在するか確認し，新たなメールが存在する場合には手順 (4) へ進み，存在しない場合には手順 (2) に戻る．
(4) 事前に登録した処理を新たなメールに対して実行する．処理の結果を受けて転送を行わない場合には手順 (3) へ戻る．
(5) 登録されている全ての転送先メールサーバにログインし，直接メールデータを保存する．
(6) 手順 (3) へ戻る．

なお，メール転送の有無にかかわらず，各処理の終了後に再び転送元 IMAP サーバに対して新たなメールが存在しないか確認を行う．これは，新しいメールの検知後に実行される処理が完了する前に，メールが新たに届いている可能性があるためである．

3.2 多数ユーザへの対応

前節で述べたように，メールデータを転送先 IMAP サーバに直接保存してメール転送を実現するには，IMAP エージェントが転送元だけでなく転送先のメールアカウント情報も把握していることが前提となる．そのため，多くのユーザが利用できる実用的なシステムとして稼働させるためには，転送に関わるアカウント情報と転送元および転送先の組み合わせをユーザ自身が編集でき，かつ IMAP エージェントがそれらの情報を取得できるような情報管理システムも必要となる．以上のことから，メール転送を担う IMAP エージェントシステムとメール転送の情報を管理するシステムの二つのシステムに分割し，**図 4** に示す構成でメール転送システムを実装する．

3.2.1 転送情報管理システム

転送時に必要なメールアカウント情報や，転送元と転送先の組み合わせをデータベースで管理する．データベースに対する情報の閲覧や追加，修正は Web インタフェースをユーザに提供することで実現する．また，ユーザに紐づいた情報に対する処理にはユーザ認証を要求する．さらに，登録済みの転送の組み合わせとは逆の組み合わせが登録されることを制限し，IMAP エージェントによる転送が無限に続いてしまう状態を回避する機能を実装する．

なお，メールアカウントのパスワードは平文で保存を行う．一般的な認証であればパスワードをハッシュ値で管理するのが妥当であるが，IMAP エージェントがユーザに代わってメールアカウントにログインする必要があるため平文のままの管理となっている．

3.2.2 IMAP エージェントシステム

転送情報管理システムに登録された情報をもとに，転送元のメールアカウントごとに，3.1 節の処理を行う IMAP エージェントのプロセスを生成する．生成したプロセスの PID（Process Identifier）は転送情報管理システムに登録することで，ユーザが Web ブラウザからプロセスの中断や再開を行うことを可能にする．また，定期的に PID テーブルを参照することで，レコードに存在するが動作していない，つまりは異常終了しているプロセスを検知し再実行することを可能とする．

© 2019 Information Processing Society of Japan

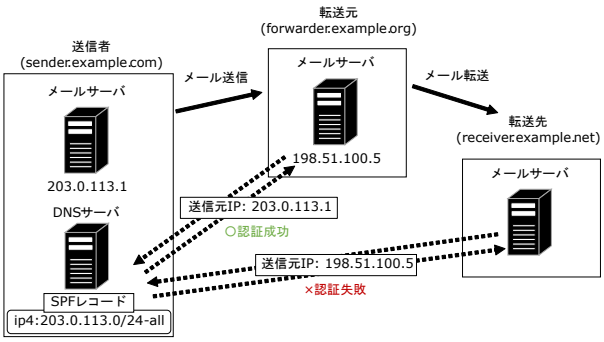

図 5 SPF 認証のメール転送問題

4. 動作検証

4.1 メール転送時の処理

転送時に実行する処理の柔軟性を示すため，送信ドメイン認証として知られる SPF 認証，DKIM 認証および DMARC 検証を転送時の処理として実装する．送信ドメイン認証では送信元のドメインに該当する DNS サーバを参照して認証処理を実行する必要があり，柔軟性を示すには十分であると考えられる．また，送信ドメイン認証に対応していない転送先に対してはセキュアな転送を実現することが可能となる．さらに，従来のメール転送において存在していた，転送先メールサーバによる SPF 認証が不当に認証に失敗する既知の問題も一部解決することができる．**図 5** に示すように，この問題は転送先のメールサーバが，転送元の IP アドレスを送信元の IP アドレスとして誤認することから生じるものである．本稿で提案するシステムでは転送先に対してメールデータを直接保存するため，この問題を回避することができる．ただし，SPF 認証で使用する送信元 IP アドレスは IMAP サーバ内のメールから直接取得することができないため，メールヘッダから検索して得られたものを利用する．具体的には，ある差出人アドレスのドメインが SMTP 通信時に使用するホストのドメイン名をあらかじめ把握しておき，メールヘッダの Recieved ヘッダから該当するドメインの IP アドレスを取得する．必ずしもこの方法で送信元 IP アドレスを取得できるとは限らないが，SPF 認証が不当に失敗していた条件下でも正当な認証結果を得られる機会を増やすことができる．

以上の送信ドメイン認証に成功した場合には検証のため，メール本文の末尾に「(Forwarded)」という文字列を追加し転送先メールサーバにメールデータの保存を行う．

4.2 検証結果

SPF 認証が正しく動作しないシナリオとして例示した図 5 と同様の問題を引き起こす環境を用意し検証を行った．なお，送信元の DNS サーバには SPF レコードに加えて DKIM の公開鍵も公開されているものとする．

説明の都合上，送信者，転送元および転送先のメールサーバをそれぞれ A，B および C とおく．A が SMTP 通信時に使用するホストのドメイン名は既知であり，情報管理システムに登録を行った．IMAP エージェントは B に対して常時接続を行い，新たなメールを検知すると DMARC の検証結果に応じて C に転送を行う．この環境で，A から検証用に DKIM-Signature ヘッダを付与したメールを B に送信したところ，IMAP エージェントにおいて以下の動作を確認することができた．

(1) 新たなメールを検知し，差出人のドメイン名をもとに送信元ホストドメインの抽出に使用する検索クエリを情報管理システムより取得
(2) Received ヘッダに対して，取得したクエリをもとに検索を行い，マッチしたヘッダ情報に記述されている IP アドレスを取得
(3) ホストドメインに対して SPF レコードを参照
(4) SPF レコードと手順 (2) の IP アドレスを比較し認証に成功
(5) DKIM-Signature ヘッダの d タグから得られるドメインから DKIM の公開鍵を取得
(6) ハッシュ値を復号後，現在のメールヘッダおよび本文から生成されたハッシュ値を比較し DKIM 認証に成功
(7) 両認証において識別子のアライメントが成立したため，DMARC 検証に成功
(8) C に接続後，対象のメールデータを APPEND コマンドにより直接保存

実際に，B に保存されたメールと IMAP エージェントが C に保存したメールを，Microsoft Outlook で確認した様子をそれぞれ**図 6** と**図 7** に示す．

転送されたメール本文の末尾には「(Forwarded)」の文字

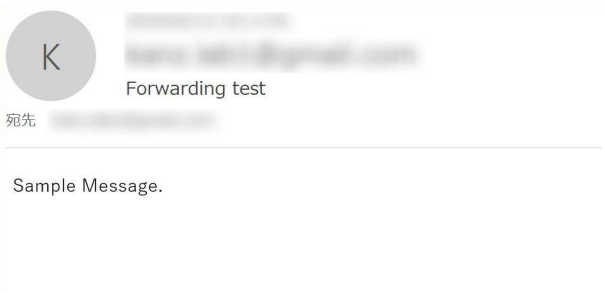

図 6 転送元の受信メール

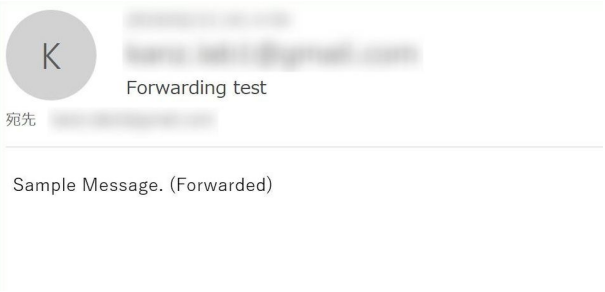

図 7 転送先の受信メール

© 2019 Information Processing Society of Japan

列が付加されていることがわかる．また，両メールのメールヘッダを比較すると完全に一致することが確認できた．したがって，限られた環境での検証にはなるが，転送時の送信ドメイン認証が正しく動作し，IMAP エージェントによる転送が行われたことを確認できた．

5. 考察

5.1 IMAP エージェントシステム

本システムでは，転送セットごとに IMAP エージェント用のプロセスを生成して並行処理を実現しているうえ，IMAP エージェントは起動されると特段の指示がない限り稼働し続けるため，ユーザの増加によってハードウェアリソースが枯渇しやすいと考えられる．また，IMAP エージェントは IMAP サーバに対する常時接続を前提としているため，ネットワークのリソースが恒常的に占有され，コネクション数が増えると常時接続が保てない事態に陥りかねない．したがって，プロセスごとにどれだけのリソースを占有するのかを考慮し，サーバ性能の向上や実行サーバの分散，転送セットの登録数に対する上限設定などが必要だと考えられる．

5.2 IMAP エージェント

実装した IMAP エージェントでは，新たなメールの検知に失敗するおそれがある．セッションを確立中はメールボックスのステータス変化を見逃すことはないが，セッションが切断された場合は再接続を行うまでの間にメールを検知できない時間が存在してしまう．セッションが切断される原因の一つとして，メールボックスのステータス更新の検知に使用している IDLE コマンドの仕様があげられる．このコマンドは接続先サーバに非アクティブと判断される可能性があり [9]，タイムアウト設定を行っているサーバからはコネクションが切断されるおそれがある．また，単純にネットワークエラーによる切断も考えられる．この問題を解決するには，IMAP サーバで各メールに紐づけられる UID（Unique Identifier）を利用してメールボックスの同期をとる必要がある．ただし，UID はセッション確立中は変化していないことが保証されているが，再接続を経ると ID が変化する可能性がある．そのため，前セッションの UID を保持して，その UID の有効性を検証する必要がある [10] ．

5.3 転送情報の管理システム

本システムでは，転送元および転送先のメールアカウント情報を平文で管理しているが，これはセキュリティ上問題のある管理方法であり，データベースの情報が抜き出された場合でも簡単にはアカウント情報が知られないよう対策する必要がある．関係データベース管理システムの MySQL を例にとれば透過的データベース暗号化 [11] 等により，システム側で特段の暗号化や復号の処理をせずにデータベースのデータファイルを暗号化することができる．ただし，SQL インジェクション等のシステムの脆弱性をついた情報漏洩には対応できないため，完全な解決とは言えず，アカウント情報の安全な管理方法については依然として実用化するうえで課題となる．

5.4 送信ドメイン認証

SPF 認証に用いる識別子の取得に事前に登録した検索条件を用いたが，これは非常に汎用性の低い方法である．検索対象となる受信メールの Received ヘッダの送信元情報の記述形式が明確に統一されていないため，あるメールサーバに対しては一致する条件でも他方のサーバでは一致しない可能性がある．また，SMTP 通信時の送信元ホストのドメインが変更された場合や大規模メールサービスではホストを通信ごとに使い分けることも考えられ，この場合にも識別子を正しく取得できないおそれがある．さらに，認証に必要な Envelope-From の取得には，IMAP サーバの FETCH コマンドを利用しているが，ここで得られる Envelepe-From の情報は SMTP 通信時の情報とは異なる [9] ため，DMARC の識別子アライメントの確認工程が意味をなさなくなる可能性がある．これに対しては，SPF の転送問題に対する解決策の一つである Envelope-From を書き換える手法のように，転送対象のメールが既に転送元で SPF 認証に成功していることを前提とするのであれば，本システムで単に転送するのみでも同様の対策になると考えられる．

また，SPF に限らず送信ドメイン認証は本来 SMTP 通信時に行われるべきものであり，DMARC でもポリシーの適用を SMTP 通信時に行うことを推奨 [7] している．そのような背景からか，メールサービスによってはメールを保存する段階で本文部に変更を加えるものも存在し，DKIM 認証に失敗する例が実際に確認できた．したがって，IMAP エージェントを使用した転送時の送信ドメイン認証は，認証タイミングの仕様上，正しく動作しない可能性がある．

6. おわりに

従来のメール転送方式において転送処理の柔軟性を高めるためにはメールサーバに追加設定が必要であった．しかし，これはメールサーバに対するユーザの権限に大きく左右されるため，外部のメールサービス等を含めて柔軟な転送処理を普遍的に実行することは困難であった．これに対して，本稿ではメールサーバに追加設定を施すことなく柔軟な転送処理を実現できる，IMAP エージェントを用いたメール転送システムを提案した．IMAP エージェントが転送元のメールサーバに常時接続を行うことで，転送元の新たなメール受信を検知し，転送先のメールサーバに直接メールを保存する方式である．加えて，実用化に向けて多

© 2019 Information Processing Society of Japan

数ユーザに対応できるよう，ユーザが転送セットを編集できる情報管理システムとその情報を利用して IMAP エージェントを並行して稼働させる IMAP エージェントシステムを実装した．転送処理の柔軟性を示す一例として実装した送信ドメイン認証を通して，その性能を示すことができ当初の目標は達成したと考えられる．また，この機能の実装により，送信ドメイン認証が抱える転送問題を解決する可能性も見出すことができた．

転送時の機能には，送信ドメイン認証が外部ネットワークと通信を行うのと同じように，外部サービスと提携した処理の実装も可能であり，他にも様々な活用方法が考えられる．依然として本システムの実用化には，スケーラビリティの低さやセキュリティの問題などの課題が残っているが，外部装置で動作する IMAP エージェントの特徴を生かした転送処理の柔軟性を発揮できており，新たな転送システムとして一考の価値があると考えられる．

参考文献

[1] Proofpoint: Open Source Email | Sendmail Sentrion| Proofpoint（オンライン），入手先 ⟨https://www.proofpoint.com/us/open-source-email-solution⟩（参照 2019-05-10）.

[2] Postfix: The Postfix Home Page（オンライン）, 入手先 ⟨http://www.postfix.org/⟩（参照 2019-05-10）.

[3] Double Precision: maildrop - mail delivery agent with filtering abilities（オンライン）, 入手先 ⟨http://www.courier-mta.org/maildrop/⟩（参照 2019-05-10）.

[4] Crispin, M: INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1, RFC3501, IETF, March 2013.

[5] 横木健太, 山井成良, 王健人, 北川直哉 : 電子メールの柔軟な処理を可能とする IMAP エージェント, マルチメディア，分散，協調とモバイル（DICOMO2018）シンポジウム, 7C-4, pp. 1403–1406, 情報処理学会, 2018 7 月.

[6] Kitterman, S.: Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1, RFC7208, IETF, April 2014.

[7] Crocker, D.: DomainKeys Identified Mail (DKIM) Signatures, RFC6376, IETF, September 2011.

[8] Kucherawy, M.: Domain-based Message Authentication, Reporting, and Conformance (DMARC), RFC7489, IETF, Match2015.

[9] Leiba, M.: IMAP4 IDLE command, RFC2177, IETF, June 1997.

[10] Melnikov, A.: Synchronization Operations for Disconnected IMAP4 Clients, RFC4549, IETF, June 2006.

[11] Oracle: MySQL Enterprise Transparent Data Encryption (TDE)（オンライン），入手先 ⟨https://www.mysql.com/jp/products/enterprise/tde.html⟩（参照 2019-05-10）.

© 2019 Information Processing Society of Japan