Vol.2018-DPS-174 No.19
Vol.2018-CSEC-80 No.19
2018/3/6

仮名化による個人情報の保護に配慮したパブリッククラウド型フィッシングメール対応訓練システムの運用と今後の展開

東野 正幸[1,a)]　川戸 聡也[1,b)]　大森 幹之[1,c)]　川村 尚生[2,d)]

概要：近年，フィッシングメールが脅威となっており，フィッシングメールに対する模擬訓練が多くの組織で実施されている．訓練を実施するための情報システムにおいては，訓練用のフィッシングサイトで訓練対象者からのアクセスログを分析し，管理画面で訓練対象者の訓練状況を把握できるシステムがあるが，訓練対象者のメールアドレス，氏名，所属等の個人に関する情報を集積している場合が多く，これは情報漏洩のリスクを高めることに繋がる．本稿では，仮名化により訓練対象者の個人に関する情報の管理主体を分離することで，システムに保存する情報量を低減し，個人情報の漏えい対策を講じることが可能な訓練システムの開発と運用について報告するとともに今後の展望を述べる．

キーワード：フィッシング攻撃，標的型攻撃，セキュリティ，セキュリティ教育

Masayuki Higashino[1,a)]　Toshiya Kawato[1,b)]　Motoyuki Ohmori[1,c)]　Takao Kawamura[2,d)]

1. はじめに

近年，フィッシングメールが脅威となっている [1], [2]．フィッシングメールとは，価値のある情報を攻撃対象者から奪い取ろうとする行動のうち，信頼されている人間やシステムに成りすまして送付される電子メールのことである．フィッシングメールによる攻撃手段の１つとして，電子メールにより攻撃対象者を偽物のウェブサイトへ誘導し偽物のログイン画面にユーザ名やパスワードを入力させることで情報システムへ不正アクセスするための情報を収集する方法がある．この攻撃の対策として，電子メールフィルタや侵入防止システムといった情報セキュリティシステムの導入に加えて，組織の構成員に対する情報リテラシ教育も重要であり，フィッシングメールに対する訓練が多くの組織で実施されている．

[1] 鳥取大学 総合メディア基盤センター
Center for Information Infrastructure & Multimedia, Tottori University, 4-101, Koyama-Minami, Tottori, Tottori 680-8550, Japan
[2] 鳥取大学大学院 工学研究科 情報エレクトロニクス専攻
Department of Information and Electronics, Graduate School of Engineering, Tottori University, 4-101, Koyama-Minami, Tottori, Tottori 680-8550, Japan
[a)] higashino@tottori-u.ac.jp
[b)] t.kawato@tottori-u.ac.jp
[c)] ohmori@tottori-u.ac.jp
[d)] kawamura@ike.tottori-u.ac.jp

攻撃対象者がフィッシングメールにより誘導されるウェブサイトを偽物であると判断するためには，ウェブサイトのドメイン名が正しいことの確認に加え，サーバ証明書も正しいことを確認する必要がある．より現実の攻撃に似た状況で訓練を実施して教育効果を高めるには，組織内よりも組織外のドメイン及びサーバ証明書を用いて訓練用のフィッシングサイトを運用することが望ましいと考えられる．

しかし，多くのオープンソースソフトウェアのフィッシングメール対応訓練システムでは，サーバに訓練対象者の氏名，役職，メールアドレス等を保存する実装になっており，組織外のサーバに個人情報を保存することは運用リスクを高めてしまう．

そこで本研究では，組織外のパブリッククラウドに配置可能なフィッシングメール対応訓練システムでありながら，訓練対象者の氏名，役職，メールアドレスといった個人情報は仮名化（pseudonymization）により管理主体を分離することで，訓練システムに係る情報セキュリティインシデント発生時に個人情報の漏洩対策を講じられるシステムを提案する．

2. 関連研究

2016 年に報告された DOGANA Project の調査 [3] によ

るとオンライン上ではフィッシング攻撃に関するツールが48件確認されている．この調査ではツールを使用目的ごとに分類し，それぞれの使用目的ごとに基準を設けてツールの評価を行っている．しかし，その評価にはツール自体のセキュリティに関する項目は含まれていない．

また，確認されているツールのうち，オープンソースソフトウェア（OSS）に該当し，かつフィッシングメール対応訓練に必要な攻撃の実行（TEAT: tools for the execution of the attack）機能と情報の集約とレポート（TIAR: tools for the information aggregation and reporting）機能の両方が利用可能なツールについて，訓練対象者の個人情報がどのように保存されているかはこの調査では明らかにされていない．このため，これらに該当するOSSのツールであるGophish[*1]，Phishing Frenzy[*2]，SPF (SpeedPhishing Framework)[*3]の機能を調査した結果，これらはサーバ上に訓練対象者の氏名，役職，メールアドレスといった識別特定情報を記録する実装となっていることが分かった．

フィッシングメール対応訓練を実施するためのツールとしてOSSを採用することは，予算が限られた組織においては実施費用の削減できる可能性があるが，訓練用のサーバに組織の構成員の氏名，役職，メールアドレスといった識別特定情報を記録することは，訓練用サーバから個人情報の漏洩リスクを高めてしまうため容易には運用できない．しかしながら，訓練用のフィッシングサイトは組織外のドメインで稼働させなければ，フィッシングサイトを見分けるための技術的な説明に説得力を持たせることが難しくなる．

一方，本研究で提案するシステムは，組織外のパブリッククラウドに配置可能なフィッシングメール対応訓練システムでありながら，訓練対象者の氏名，役職，メールアドレスといった識別特定情報は仮名化により管理主体を分離することで，訓練システムに係る情報セキュリティインシデント発生時に個人情報の漏洩対策を講じられるシステムである点で従来のシステムとは異なり，サーバの運用や情報セキュリティ対策に関する高度な技術は多くを要求しない．

3. システムの設計

訓練の実施状況を把握するためには，訓練対象者が訓練用のフィッシングサイトに情報を入力したかどうかを調査する必要がある．ヒアリングやアンケートなどによる調査の場合，フィッシングサイトに引っかかる人はなぜ引っかかったのか把握できていない場合があり正確性の担保が難しい．また，将来的に組織の構成員全員に訓練を実施することが想定される場合には，正確性の担保の難しさだけでなく，調査に要する時間が膨大となる．このため，フィッシングメール対応訓練を安価に実施するためには訓練対象者の行動を自動的に記録する必要がある．

しかしながら，既存のオープンソースソフトウェアは，訓練対象者のメールアドレスやパスワードをサーバ上のデータベースに保存するタイプがほとんどであり，これを組織外のサーバで運用した場合，個人情報の漏洩対策が必要となり，情報セキュリティにおけるコストとリスクが高まる．

そこで提案システムでは，仮名化（pseudonymization）により，識別特定情報と非識別非特定情報に情報を分離し，後者のみをサーバに記録する．ここで，識別特定情報とは「個人が（識別されかつ）特定される状態の情報（それが誰か一人の情報であることがわかり，さらに，その一人が誰であるかがわかる情報）」である．非識別非特定情報とは「一人ひとりが識別されない（かつ個人が特定されない）状態の情報（それが誰の情報であるかがわからず，さらに，それが誰か一人の情報であることが分からない情報）」である．これらの定義は文献[4]に基づく．仮名化の導入により，もしサーバから情報が漏洩した場合であっても，分離して保管されている識別特定情報を削除することで，サーバから漏洩した情報から個人を特定することが難しくなる．

4. システムの実装

提案システムはサーバ・アプリケーションとクライアント・アプリケーションの2つで構成される．サーバ・アプリケーションは訓練対象者の振る舞いを記録する機能を持つ．クライアント・アプリケーションは訓練用サイトへ誘導するメールを送付する機能を持つ．

4.1 サーバ・アプリケーションの実装

訓練用サイトと訓練対象者の振る舞いを記録するサーバ・アプリケーションはRuby on Rails[*4]で実装した．

4.1.1 訓練対象者の行動記録

本サーバには，タイムスタンプ，訓練ID（Campaign ID），訓練対象者ID（Person ID），及び訓練対象者のアクションが記録される（**図1**，**図2**）．

アクションには，メールの開封（Opened E-mail），リンクのクリック（Clicked Link），情報の送信（Submitted Data），訓練結果画面の閲覧（Viewed Result）の4種類を定義する．それぞれのアクションの取得方法は下記のとおりである．

[*1] Gophish: `https://getgophish.com/` (accessed 2017-08-25).
[*2] Phishing Frenzy: `https://www.phishingfrenzy.com/` (accessed 2017-08-25).
[*3] SPF (SpeedPhishing Framework): `https://github.com/tatanus/SPF` (accessed 2017-08-25).
[*4] Ruby on Rails — A web-application framework that includes everything needed to create database-backed web applications according to the Model-View-Controller (MVC) pattern.: `http://rubyonrails.org/` (accessed 2017-08-25).

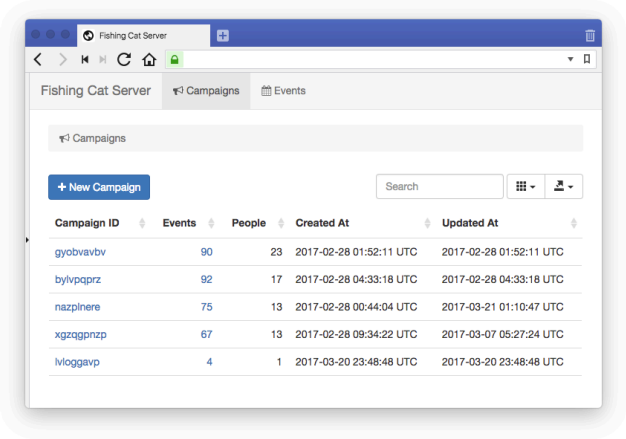

図 1　訓練一覧画面

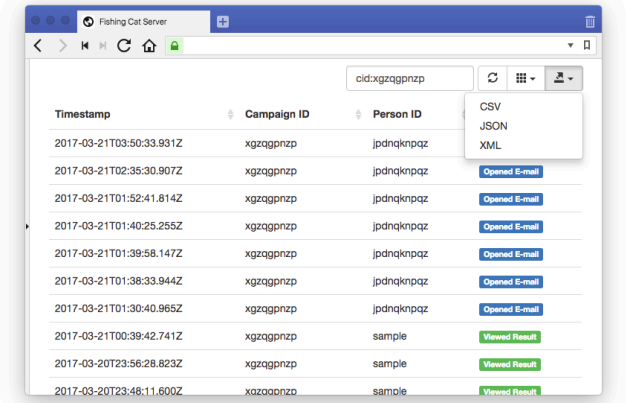

図 2　イベント管理画面

Opened E-mail　HTML メールに含める img タグの src 要素にサーバの URL を記述する．HTML メールを開封すると画像が表示され，その際にアクセスした URL により訓練対象者によるメール開封を判定する．URL には訓練対象者 ID を含めているため，誰がメールを開封したかを識別できる．

Clicked Link　訓練メールに記載した URL に HTTP による GET メソッドでアクセスした場合に記録される．アクセス先のウェブページにはウェブフォームを設置する．

Submitted Data　訓練メールに記載した URL に HTTP による POST メソッドでアクセスした場合に記録される．

Viewed Result　HTTP による POST メソッドでアクセスした後にリダイレクトさせる URL にアクセスした場合に記録される．本ウェブサイトが訓練用サイトであることの説明を記載する．

上記のアクションは**表 1** に示す URL により表現する．URL に含まれる `cid` は訓練 ID（Campaign ID），`pid` は訓練対象者 ID（Person ID）を表す．

以上の情報には，訓練対象者の氏名やメールアドレスといった識別特定情報は含まれない．その代わりに，識別特定情報に依存せず生成された訓練対象者 ID（Person ID）を用いて仮名化データ（pseudonymized data）として保存する．これにより，標的型メール攻撃訓練を組織外のサーバで実施しながら，訓練用サイトにおける情報漏洩のリスクを低減させることが可能となる．訓練対象者 ID の生成は，後述するクライアント・アプリケーションで訓練メールを送付した際に生成するため，訓練対象者 ID とメールアドレスの紐付けはサーバでは無くローカルでのみ実施する．

表 1　各アクションのルーティング

Action	Method	Path
Opened E-mail	`GET`	`/images/:cid/:pid`
Clicked Link	`GET`	`/forms/:cid/:pid`
Submitted Data	`POST`	`/forms/:cid/:pid`
Viewed Result	`GET`	`/results/:cid/:pid`

この様に，訓練用サーバの運用組織と，訓練用メールの送信組織を分離可能にすることで，仮に訓練用サーバに情報セキュリティインシデントが生じた場合には，情報が分離されているため訓練対象者の識別及び特定を困難にできる．このため，安価にかつ効果的な標的型攻撃訓練を実施しながらも，訓練用サーバにかかる情報セキュリティインシデントのリスクを低減することが可能となる．

4.1.2　訓練実施担当者向け管理機能

訓練用のフィッシングサイトのウェブページのデザインは，訓練を行う組織の構成員が良く使用するデザインと同一のものにすることが望ましい．また，訓練結果のウェブページも訓練であることを説明や訓練を担当する部署の連絡先などを記述できる必要がある．そこで，機密情報の詐取と訓練結果の説明を行うためのウェブページは，訓練の実施担当者が管理画面により任意の HTML 及び CSS により変更できるようにした（**図 3**）．

これらの変更機能により作成したウェブページの作成例を図 3 及び図 3 に示す．本機能によるウェブページの作成には Ruby on Rails に標準で組み込まれている ERB (Embedded Ruby) のテンプレートエンジンを使用することができる．また，テキストフィールドやパスワードフィールドは Ruby on Rails に標準で組み込まれているヘルパーメソッドである `text_field_tag` や `password_field_tag` を用いることで容易に実装できる．また，このヘルパーメソッドを使用してテキストフィールドやパスワードフィールドを設置することで，実際にはテキストフィールドの情報を送信しないように設定することも可能にしている．これにより，訓練中に正規のユーザ ID やパスワードを送信してしまうことによる情報漏洩を防ぐことができる．

4.2　クライアント・アプリケーションの実装

訓練メールを送信するためのクライアント・アプリケー

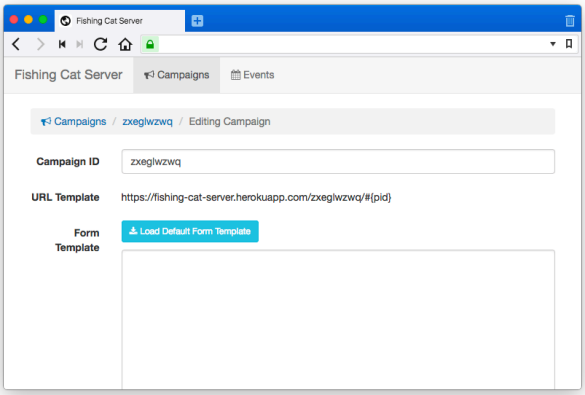

図 3　訓練用ウェブページ及び訓練説明用ウェブページの編集画面

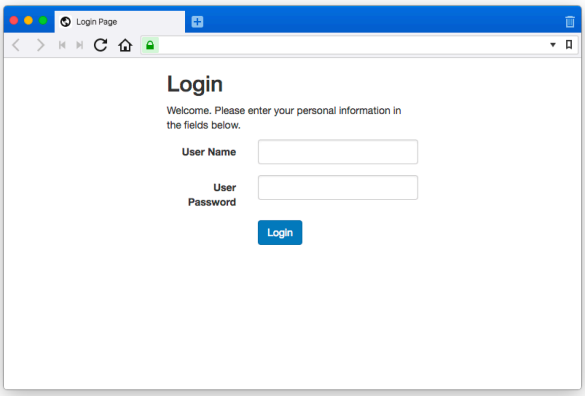

図 4　訓練用ウェブページのサンプル

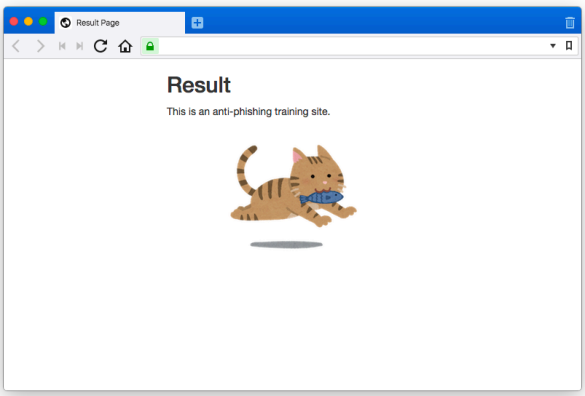

図 5　訓練結果表示用ウェブページのサンプル

ションは Ruby[*5]で実装した．訓練メールの本文は ERB（Embedded Ruby）によるテンプレートエンジンによりプレーンテキスト形式と HTML 形式のメールを生成できる．メールの送信には ActionMailer を使用した．ActionMailer は Ruby on Rails にも標準で組み込まれており Ruby のメール送信ライブラリとして使用実績が多数存在する．訓練メールの送信時にはそれぞれのメールアドレスに対して全く関係しないランダムな文字列を訓練対象者 ID として生成及び付与し，その訓練対象者 ID をメールの本文中の URL に含めることで，訓練対象者のアクションを追跡するようにした．ランダムな文字列の生成には hashids[*6]を使用した．

また，フィッシングメール対策訓練は継続的に実施することでフィッシングメールに対する対応方法の変化を定量的に評価し対策を講じることが有効と考えられる．このため，訓練をある程度は自動的に実施できるインタフェースにすることが望ましい．そこで，本ツールはコマンドラインツールとして実装し，crontab などのコマンドの定時実行スケジュール管理ツール等と組み合わせて使用することで継続的かつ自動的な訓練の実施にも対応できるようにした．

5. システムの試運用

本システムの試運用のため本学の職員 35 名を対象に訓練を実施した．

5.1 訓練概要

本学では統一認証基盤により様々なサービスのシングルサインオンを実現している．また，学外からのアクセスに対しては多要素認証等の導入により安全性を確保している．統一認証のユーザ ID とパスワードが漏洩した場合には，機密性の高い情報のさらなる流出に繋がる可能性があるため，事前の訓練により教育・啓蒙を実施することで被害回避能力を組織的に高めることは重要である．そこで，標的型メール攻撃により統一認証のユーザ ID とパスワードが漏洩する状況を想定した訓練を実施した．本訓練は，訓練対象者に標的型メール攻撃を模倣した HTML メールを送付することで，訓練対象者を統一認証のログイン画面に模倣した訓練用のフィッシングサイトへ誘導し，統一認証のユーザ ID 及びパスワードを訓練用のフィッシングサイトのウェブフォームで送信させることで，機密情報が漏洩する状況を想定して実施した．実施期間は 2017 年 3 月 21 日（火）から 2017 年 3 月 31 日（金）までの 11 日間とした．訓練対象者は本学の職員 35 名とした．

5.2 サーバ・アプリケーションの運用

サーバ・アプリケーションの運用には Heroku[*7]の無料プランを使用した．無料プランでは 30 秒間アクセスがない場合にアプリケーションのインスタンスが停止する制限（再度アクセスすればインスタンスは自動的に復帰するが

*5 Ruby Programming Language: https://www.ruby-lang.org/ (accessed 2017-08-25).

*6 Hashids - generate short unique ids from integers: http://hashids.org/(accessed2017-08-25).

*7 Heroku: Cloud Application Platform https://www.heroku.com (accessed 2017-08-25).

停止から起動のための時間がかかり停止時の初回アクセスのレスポンスが低下する.）や，1 ヶ月あたりの稼働時間の制限や，データベースに PostgreSQL を使用した場合にレコードの行数が 10,000 行までの制限がある。本システムでは 4 種類のアクションを記録することから，全員が全てのアクションを 1 回ずつ実行したとして無料プランでは最大で 2,500 人分まで記録できる．ただし，何度もアクションを実行する訓練対象者もいることや，全員がアクションを実行することはあまりないことから，1,000 人程度までであれば無料プランで実施可能であると考えられる．

5.3 クライアント・アプリケーションの運用

クライアント・アプリケーションによる訓練メールを送信するためのメールサーバには GMO インターネット株式会社のクラウドサービスである ConoHa[*8]を採用した．ConoHa は数クリックでメールサーバを配置可能であり，メールサーバを利用する場合の費用は日本（東京）リージョンのサーバでは 2017 年 4 月 17 日現在で 1 時間あたり 1.7 円となっている．このため，訓練メールを送信した後にメールサーバを削除することで，メールサーバ費用は数円程度で実施可能となる上，メールアドレスも自由に作成可能であり，訓練の柔軟性においても利点がある．

訓練メールの文面作成においては，最近の電子メールクライアントでは HTML メールにおいて<a>タグの href 属性の URL とコンテンツの表記が異なる場合に詐欺メールの可能性を指摘する警告が示される場合がある．正規の URL を<a>タグにより偽装する方法はユーザへ明示的に警告が示される場合があるため，攻撃手段として有効性が下がる可能性がある．このため，あえて訓練用ウェブページの URL は外部サーバのドメインが分かるようにし，偽装を行わないこととした．

訓練対象者に送付する訓練メールの送信においては，電子メールの本文へ訓練対象者を一意に識別する pid を含む URL を付与するため，同じ内容の電子メールを同報送信することができない．このため，1 通ずつ電子メールを順番に送付を行う必要があるが，短時間での多量の電子メールの送信は，メールサーバの負荷対策機能により行えない場合がある．そこで，本訓練では電子メールを 1 通送信するごとに 5 秒のインターバルを加えた．もし，訓練対象者が数千人に及び，なおかつ短期間で訓練を完了させなければいけない場合は，複数のメールサーバを使用して送信する方法などの検討が必要になる．

5.4 訓練結果

訓練対象者 35 名に本のうち 7 名が何らかのアクションを行った．**表 2** に訓練結果を示す．うち 1 名はリンクのクリックのみにとどまり情報の送信は行わなかった．その他の 6 名は情報の送信を行い，17.1%にあたる訓練対象者が情報を送信した結果となった．

本訓練では個々の訓練対象者に 1 通のみ訓練メールを送付したが，より訓練の効果を高めるためには，何通かのメールを継続的に送付することで，フィッシングメールに対する対応方法の改善効果を定量的に評価する必要があると考えられる．

表 2 訓練結果

番号	ページの閲覧	データの送信	担当者への連絡
1	✓	✓	✓
2	✓	✓	✓
3	✓	✓	✓
4	✓	✓	
5	✓	✓	
6	✓	✓	
7	✓		

5.5 訓練におけるシステムの評価

本システムにより訓練を実施する前では，訓練対象者のふるまいの傾向を把握できれば十分であると考えていた．しかし，実際に運用を終えると，組織内で実施される情報セキュリティに関する講習会の参加記録や，実際に発生した情報セキュリティインシデントの記録などと組み合わせることで，情報セキュリティに関する講習会の内容設計や，情報セキュリティインシデントの予防方策の決定などに活用できる可能性が示唆された．

このことから，本システムは他の事業との連携においても有効活用できる可能性があり，本システムに保存されたデータの柔軟な検索機能やエクスポート機能が必要であると考えられる．

6. おわりに

本システムは，標的型メール攻撃やフィッシング攻撃の対応訓練を効果的かつ安全に，そして安価に実施可能な設計になっている．本システムのソースコードは https://github.com/fishing-cat/ で公開している．ただし，現在は一般利用において必要なマニュアルの整備やエラー時の処理の実装が未完のまま残っている．今後は，本学以外の組織においても活用できるようにシステムとマニュアルを整備しその有用性を評価していく．

また，数千人を対象とした大規模訓練も実施済みであり，システムの性能的な有用性は確認できている．また，訓練対象者にヒアリングやアンケートなどを実施し，情報セキュリティインシデント発生時の対応方法が十分に認識されているか調査を実施する予定である．さらに今後は，サーバに保存される情報の匿名性の評価を進めるととも

[*8] ConoHa: https://www.conoha.jp/ (accessed 2017-08-25).

に，複数の組織が共同利用することにより個々の組織の知見を互助的に活用することで効果的な訓練を実施可能なシステムへの展開を模索したい．

参考文献

[1] Anti-Phishing Working Group, Inc. (APWG): Phishing Activity Trends Report 4th Quarter 2016 (2017).
[2] フィッシング対策協議会ガイドライン策定ワーキンググループ：フィッシングレポート 2016 ―世界に広がるフィッシング対策の輪 ― (2016).
[3] Dambra, C., Gralewski, A., Frumento, E., Puricelli, R., Valentini, F., Mamelli, A., Russo, M., Weiss, N., Pacheco, B., Segou, O., Beaume, J. and Custodio, F.: Report on existing tools, their evaluation and the gap to be filled by DOGANA development, *Advanced Social Engineering and Vulnerability Assessment Framework*, DOGANA Project (2016).
[4] 技術検討ワーキンググループ：【資料 2-1】技術検討ワーキンググループ報告書，第 5 回パーソナルデータに関する検討会 議事次第 (2013).