短縮 URL の安全な利用に向けて

Towards Safe Use of Short URLs

○中村 章人[1]，松尾 卓朗[1]，林 隆史[1]

Akihito NAKAMURA, Takurou MATSUO, Takafumi HAYASHI

1 会津大学 コンピュータ理工学部　School of Computer Science and Engineering, University of Aizu

Abstract　Cyber attack is one of the most serious threats facing many organizations in the Internet era. In addition to hardening computer and network systems, it is important to surely deliver security information to end users. This paper presents a method and system for safe-browsing, especially of shortened URL links. Short URLs may be utilized for malicious activities to redirect users to unexpected resources, e.g. phishing and malware, by obscuring the final destinations. The proposed method enables users to know how safe a particular Web resource might be before users dereference it. Our system retrieves and delivers safety information of the long URLs on user's demand by a simple operation.

キーワード　情報セキュリティ、Web、短縮 URL、フィッシング

１．はじめに

短縮URLは、インターネット利用者の利便性を高める便利なしくみである。しかし、その最終的なアクセス先が利用者から隠蔽されてしまうため、フィッシングやマルウェア感染を目的とした悪意あるサイトへの誘導など、セキュリティ上のリスクがある。

短縮URLの安全性を検査するサービスがいくつかあるが、いずれもタイピングまたはコピー&ペーストを用いてHTMLフォームに検査対象のURLを入力する必要がある[6,7]。すなわち、利用者は表示中のWebページやソーシャルネットワーキングサービス（SNS: Social Networking Service）の投稿記事からいったん検査サービスのWebページに画面遷移しなければならない。この手間は、利用者が安全性検査を回避してしまう原因となる。

本論文では、ユーザビリティを損ねることなしに、短縮URLを安全に利用するしくみを提案する。オンデマンドで最終アクセス先リソースの安全性を検査し、ユーザに評価情報を提示する。

提案手法では、利用者はWebページに表示された短縮URL上でマウスクリックして表示されるコンテキストメニューを選択するだけで済む。1 ステップで検査を実施して評価情報に到達でき、中間の画面遷移やURL入力のステップはない。また、サーバ・クライアント型システムを採用することで、サーバ側での評価情報のキャッシュや統計データの取得を可能にしている。また、再帰的に複数回の短縮をされたURLに対して、複数リダイレクトを制限するWebブラウザの安全機能に関係なく検査できる。

本論文の構成は次のとおりである。2章では、短縮URLおよびURL短縮サービスの動作原理を説明する。3章では、我々が提案するURLの安全性検査手法とそれを実現するシステムについて述べる。4章で今後の課題を示し、5章で結論を述べる。

２．短縮 URL と URL 短縮サービス

本章では、短縮URLとURL短縮システムについて説明する。URL短縮システムとは、あるURLから短縮URLを生成し、短縮URLから元のURLへの逆変換を行うシステムである。このシステムをWebサービス化したものがURL短縮サービスである。

２．１．短縮 URL

短縮URLとは、あるURL（元URL）に対してその長さを短くしたURLである。短いURLは、SNSで投稿記事の長さが制限されている場合に文字数の消費を抑える、Webページの見栄えを損ねないなどの利点がある。**表 1**に、主なURL短縮サービスと短縮URLの例を示す。ここでは、任意のURLを短縮できるサービスのみを選んだ。

短縮URLは、URL短縮サービスのトップレベルドメインと、元URLに対応する一意なキーとから構成される。キーは、アルファベットの大文字および小文字と数字の合計62文字、すなわちBase62エンコーディングを使用するものが多い。同一の元URLに対して生成される短縮URLはサービスごとに異なり、短縮URLから元URLを復元できるのはその短縮を行ったサービスだけである。

表 1に示した例のように、短縮URLから元URLを推測することは困難である。すなわち、短縮URLは最終アクセス先を利用者から隠蔽してしまうため、攻撃に悪用されるリスクがある。

表 1：主な URL 短縮サービスと短縮 URL の例

元 URL：https://en.wikipedia.org/wiki/Japan

サービス提供者	短縮URL
Bitly	http://bit.ly/1LXn5Yl
アスキー・メディアワークス	http://go.ascii.jp/hn2
Google	https://goo.gl/J0HsVM
HootSuite	http://ow.ly/EkMY301qA1p
TinyURL	http://tinyurl.com/65baxhe

２．２．URL 短縮サービスの動作原理

URL短縮サービスは、短縮URLを作成し、短縮URLへのアクセスに対して元URLへのリダイレクトを行うサービスである。主な構成要素は、短縮URLの作成アルゴリズムと、短縮URLと元URLとの対応表である。以下にURL短縮サービスの基本的な動作を示す（図1）。

URL 短縮サービスの動作:

（１）URL提供者Xは、URL_Lで指示されるWebリソースAに対して短縮URLを作成することを決める。
（２）XはURL短縮サービスPに元URL（URL_L）を送り、短縮URLの作成を依頼する。
（３）Pは、URL_Lにあるアルゴリズムを適用して短縮URL（URL_S）を得る。
（４）P はURL_LとURL_Sの組をURL対応表に登録する。
（５）P はURL_SをXに送る。
（６）Xはあるリソース B にURL_Sを掲載する。
（７）URL利用者Yは、Bとそれに含まれるURL_Sを購読する。
（８）YはURL_Sで指示されたリソースへのアクセスを試みる。実際にはWebブラウザなどのユーザエージェントがURL_Sを解釈してPにHTTP GET要求を送る。
（９）P はURL_Sを伸長するために、対応表からURL_Sを検索してURL_Lを得る。
（１０）P はURL_LをYに送る。実際にはURL_Lを含むHTTP 301応答（Moved Permanently）をユーザエージェントに返す。
（１１）YはURL_Lで指示されたリソースへのアクセスを試みる。実際にはユーザエージェントがリダイレクト処理を行ってAを取得する。

短縮URLから元URLへの逆変換は、短縮を行ったサービスだけができる。しかし、利用者にとっては一つのサービスで任意の短縮URLを逆変換できる方が望ましい。このようなサービスを複合型URL伸張サービスと呼ぶ。具体的には、ExpandURL[6]や短縮URLチェッカー[7]などがこれにあたる。

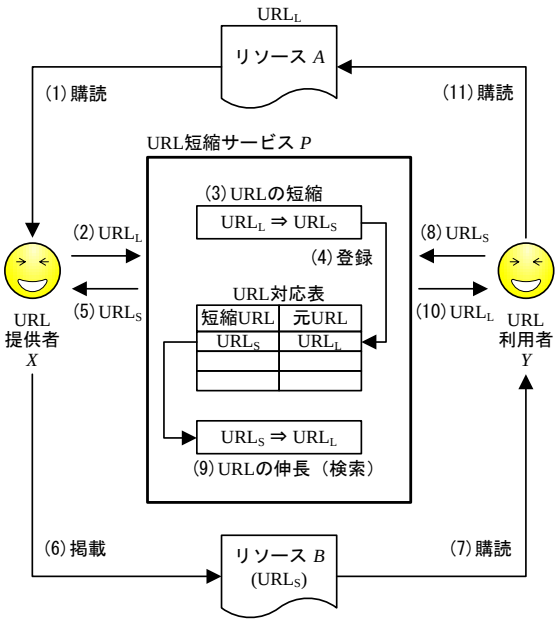

図 1：URL短縮サービスの動作原理

２．３．既存の URL 安全性検査サービス

URLの安全性を検査するサービスには2種類のものがある。一つは、一般的なURLに対する検査を行うもので、主にセキュリティベンダーが提供している。この種のサービスをWebセキュリティ評価（または評判）サービスと呼ぶ。

もう一つは、任意の短縮URLから元URLへの変換を行うURL伸張サービスが、付加的な機能として検査を行うものである。実際には、前者のWebセキュリティ評価サービスに検査を委譲している。

これらのサービスでは、短縮を再帰的に多重に行って得られた短縮URLを完全に伸長できない、日本語文字コードを正しく扱えないなどの問題がある。また、検査結果を文字情報だけで示す場合、利用者は理解しにくいという課題がある。

３．提案手法: URL のオンデマンド安全性検査

本章では、我々が提案するURLの安全性検査手法と、それを実現するシステムについて述べる。

３．１．安全性検査の手順とシステム構成

まず、URL安全性検査の手順を示す（図 2）。

URL 安全性検査の手順:

（１）URL安全性検査クライアントCは、短縮URL（URL_S）をサーバに送る。
（２）URL安全性検査サーバSは、URL_Sのトップレベルドメインに対応する外部URL短縮サービスを呼び出し、伸長結果の元URL（URL_L）を得る。

URL_Sが多重に短縮されている場合は、伸張処理を繰り返す。キャッシュに検査結果が存在する場合は、その検査結果をCに送り、以下の処理を省略する。

（３）URL_Lの安全性検査を外部の検査サービス（設定により複数）に委譲し、検査結果（$R(URL_S)$）を作成する。

（４）$R(URL_S)$をキャッシュに保存する。

（５）$R(URL_S)$をクライアントに送る。

（６）Cは、サーバから検査結果$R(URL_S)$を受け取り、それを別の画面（ウィンドウやタブ、またはポップアップ）に表示する。

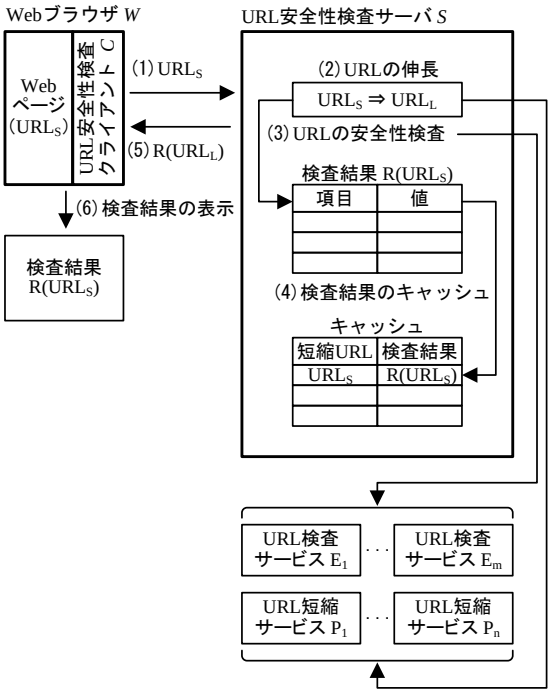

図 2：URL安全性検査の手順

３．２．ユーザインタフェース

我々が開発したシステムのユーザインタフェース（UI）について説明する。UIについて重要視した要件は、利用者のユーザビリティを損ねないことである。利用者に要求する操作が難しかったり煩雑であったりしては、この安全性検査機能が利用されない。これからアクセスしようとするWebサイトなどの安全性を確かめず、すぐにそのリソースにアクセスしてしまう。本システムでは、URLの入力やそのための画面遷移など、不要な操作ステップを極力排除し、１ステップで検査結果に到達できるようにした。

以下に利用者の操作手順を示す。

利用者の操作手順:

（１）Webブラウザ上で、短縮URLの上でマウス右クリックする。

（２）コンテキストメニューが表示されるので、短縮URL安全性検査機能を選択する。メニューには『短縮URL「実際の短縮URL」を検査』と表示される。[**図 3**]

（３）別のWebページが開き、評価情報が表示される。これを参考に、元のページで短縮URLのリソースにアクセスするかどうか判断する。

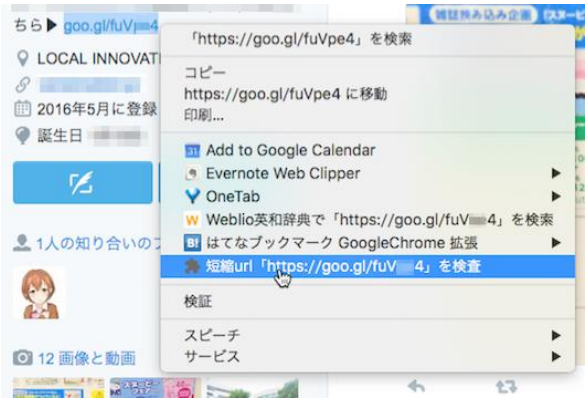

図 3：URL 安全性検査のユーザインタフェース

既存のURL検査システムでは、利用者はまずURL安全性検査サービスが提供するWebページを開き、対象のURLをタイピングやコピー＆ペーストでHTMLフォームに入力しなければならない。これに対し、我々のシステムでは、利用者は画面遷移やURLの入力を必要とせず、画面上のURLに対する直接操作ができる。このことから、既存システムのUIに比べてユーザビリティが高いといえる。

３．３．検査結果情報

URL安全性検査サーバがクライアントに返す検査結果は以下の項目から構成される。

URL 安全性検査結果の内容:

- 短縮URL情報
  - 短縮URL
  - 作成日時
  - 元URL
  - 多重短縮の場合、伸長過程（短縮URLと元URLのリスト）
- 元URLのリソース情報
  - MIMEタイプ
  - 文字コード
  - タイトルなどのコンテンツ情報

- サムネール画像
- 利用者情報
  - リファラ
  - アクセス数（伸長回数）
  - アクセス元地域

検査結果情報の表示画面は現在開発中であるため、一部のみを**図 4** に示す。ここでは Safe Browsing APIs (v4)[9]で得られる結果を利用している。これ以外にも、セキュリティベンダーが提供している Web セキュリティ評価サービスの結果や悪性サイトのブラックリストなどの情報を統合する予定である。

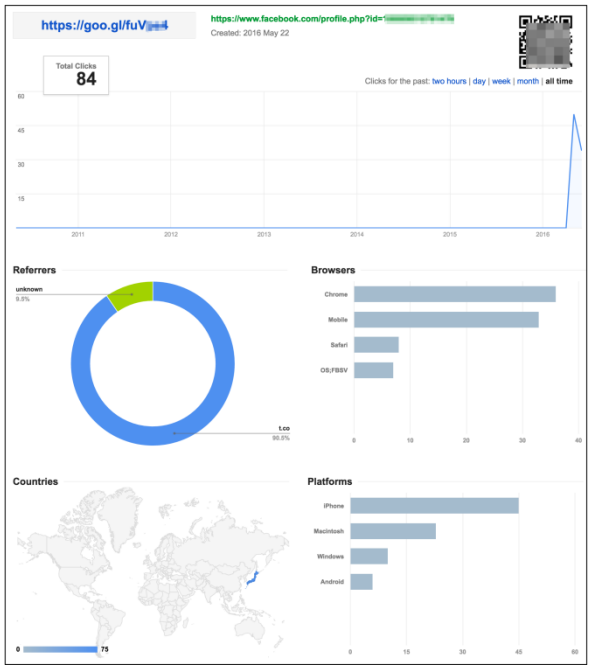

図 4: URL 安全性検査結果の例

３．４．システムの実装

提案手法に基づくシステムの基本的な機能は実装済みである。既に述べたように、クライアント・サーバ構成にした。実行環境は、まず利用者数の多いブラウザChromeを対象とし、他のブラウザへの対応は今後の課題とした。サーバ側はRuby on Railsを使用してRuby言語で、クライアント側はChrome Extensions[10]を使用してJavaScriptで開発した。また、サーバ側のデータベースにPostgreSQLを、外部サービスへのアクセスにRuby gemのcapybaraを用いた。

外部サービスの利用状況は次のとおりである。まず、URL短縮サービスは**表 1**に示したものを利用している。また、URLの安全性検査にはGoogle Safe Browsing APIs (v4) [8, 9]を用いた。

４．今後の課題

本章では、提案手法の改善すべき点と今後の研究課題について述べる。

検査結果キャッシュのタイムアウト時間の設定

この時間を適切に設定するのは難しい。短縮URLの作成が最近であるほど、タイムアウト時間を短くするのがよいと考える。なぜならば、新しい情報ほど急速に広がる可能性があるからである。古い短縮URLは、対応する元URLも古いので、検査結果の変化は少ないと予想される。

一方、悪意あるサイトの作成者は、短時間で次々と誘導情報（短縮URLを含む）および攻撃用サイトを変更する傾向がある。攻撃用サイトの生存期間の測定手法[4]などを考慮した検査情報の作成が必要であろう。

スクレーピングの高速化

クライアントから見た応答性を高めるために、URLの評価結果をサーバ側でキャッシュする工夫をしている。加えて、サーバのURL評価にかかる処理時間も短くしたい。そのためには、外部サービスへのアクセス、すなわちスクレーピングを非同期・平行実行するのが有効である。しかし、多重短縮URLの伸長は並列化できない上、短縮URLの伸長が終わってからでなければ安全性検査サービスの呼び出しができない。このような状況で、サーバ側での処理時間の短縮を工夫したい。

サーバログからの知見の獲得

サーバではURL検査結果のログだけでなく、クライアントからの検査要求もログに記録する。このログを分析することで、URLの安全性について何らかの知見が得られるものと考えている。

例えば、危険な短縮URLの特徴は、安全性検査機能を持つURL短縮サービスを使用せずかつ多重短縮されている、広く利用されているURL短縮サービスではなく独自のものを使用しておりそのトップレベルドメインはブラックリストに含まれている、といった仮説が裏付けられるかを確かめたい。そのためには、本サービスを一般に公開し、相当数の利用者を獲得する必要がある。

安全性検査結果に基づく利用者行動の抑制

ブラウザの警告に対する利用者行動の大規模な調査研究[1]によれば、2種類のブラウザ（Mozilla FirefoxとGoogle Chrome）でそれぞれ9.1%および18.0%の利用者がフィッシングサイトの警告を無視していた。また、マルウェアの警告に関してはそれぞれ7.2%と23.2%の利用者が無視していた。この問題は、技術的な対策だけでは対応できない。セキュリティに対する人の心理的・行動的特性に関する研究[2,3,5]が必要であろう。

５．結論

本論文では、短縮URLのリスクに着目し、オンデマンドでそのリンク先リソースの安全を検査するしくみを示した。一般利用者のユーザビリティを考慮して、ブラウザ上での簡単かつ直接操作により検査結果を表示できる点が特徴である。提案手法はクライアント・サーバ構成のシステムを実装済みで、Google Chromeなどの一般的なブラウザで動作する。

今後は、検査手法とシステム実装の改善を図るとともに、検査結果の表示のしかたの工夫、およびログの分析に取り組みたい。

参考文献

[1] Akhawe, D., Felt, A.P. (2013): “Alice in Warningland: A Large-Scale Field Study of Browser Security Warning Effectiveness”, 22nd USENIX Security Symposium, 2013.

[2] 西岡大, 齊藤義仰, 村山優子: “専門知識のないユーザを対象とした情報セキュリティ技術に関する安心感の構造”, 情報処理学会論文誌, Vol.54, No.9, 2013, pp.2197-2207.

[3] 寺田剛陽, 津田宏, 片山佳則, 鳥居悟: “IT被害に遭いやすい心理的・行動的特性に関する調査”, マルチメディア、情報処理学会 分散協調とモバイルシンポジウム2014論文集, 2014, pp.1498-1505.

[4] 秋山満昭, 八木毅, 針生剛男: “改ざんWebサイトのリダイレクトに基づく悪性Webサイトの生存期間測定”, 電子情報通信学会 信学技報, ICSS2013-71 (2014-3), 2014.

[5] 寺田剛陽, 鳥居悟, 安野智子, 瀧澤弘和, 新真知: “リスク認知に基づく標的型メール対策の検討”, 情報処理学会研究報告セキュリティ心理学とトラスト, Vol.2013-SPT-5, No.9, 2013.

[6] ExpandURL, http://www.expandurl.net/

[7] 短縮URLチェッカー, http://x-1.jp/

[8] Google 透明性レポート セーフブラウジングのサイトステータス, https://www.google.com/transparencyreport/safebrowsing/diagnostic/

[9] Google Safe Browsing APIs (v4), https://developers.google.com/safe-browsing/v4/

[10] Google Chrome Extensions, https://developer.chrome.com/extensions