6S-08

ブラウザ部分仮想化の提案

小泉 修一† 早川 智一‡ 疋田 輝雄‡
明治大学大学院 理工学研究科† 明治大学 理工学部‡

1. はじめに

WWW（World Wide Web）の普及に伴い，悪意のあるコンテンツを含む Web ページ（以下，悪意のあるページ）から閲覧者を保護する手法が必要となってきている．たとえば，悪意のあるページを利用する攻撃としては，ドライブ・バイ・ダウンロード攻撃や Image Gate 攻撃があげられる．

この保護手法の 1 つとして，先行研究（2 節）で提案された「仮想 Web ブラウザ」（以下，仮想ブラウザ）がある．仮想ブラウザは，Web ページ（以下，ページ）全体を透過的に画像化して悪意のあるページを無害化する．

しかし，仮想ブラウザには，生のブラウザ（例：Chrome や Firefox）と比較してユーザビリティ（応答性・再現性）が低下しやすいという課題がある．これは，仮想ブラウザが，安全性を優先して本来は不要な箇所まで画像化しているためである．これにより，画像化に必要な時間が増加して応答性が低下し，悪意のない動的コンテンツ（例：動画）も画像化されて動作しなくなり再現性が低下する．

本稿では，仮想ブラウザのユーザビリティを向上させるために，部分的な仮想化を提案する．具体的には，ページ全体を画像化するのではなく，閲覧者にとって脅威となりうる箇所（以下，危険箇所）のみを画像化する．危険箇所の決定には，安全性と利便性とのトレードオフを念頭に置く必要がある．そこで，本稿では，1 つの切り口として，オリジンと HTML タグとを用いる手法を採用した（3 節）．我々は，提案手法のプロトタイプを設計（4 節）・実装（5 節）して評価（6 節）を行い，提案手法が仮想ブラウザのユーザビリティの向上に寄与するという結論を得た．

2. 先行研究

早川ら[1]は，HTML5 と JavaScript 関連技術とを用いた仮想ブラウザを提案した．仮想ブラウザは，サーバ群（「認証・中継サーバ」と「画像化サーバ」）と専用クライアントで構成される．閲覧者がページをリクエストすると，（1）認証・中継サーバがリクエストを画像化サーバに転送し，（2）画像化サーバが対象ページを画像化し，（3）専用クライアントが画像化されたページを表示する．

Partial virtualization for browsers

†Shuichi Koizumi, ‡Tomokazu Hayakawa, ‡Teruo Hikita
†The Graduate School of Science and Engineering, Meiji University
‡School of Science and Technology, Meiji University

3. 提案手法

我々は，ページ中の危険箇所のみを判別して画像化する部分仮想化を提案する．具体的には，オリジン（3.1 節）と HTML タグ（3.2 節）とを用いて危険箇所を判別する．これにより，画像化の範囲を制限できるため，画像化に必要な時間の低減（応答性の向上）と悪意のない動的コンテンツの動作（再現性の向上）とが期待できる．

部分仮想化の実行例を図 1 に示す（情報処理学会第 79 回全国大会のページ[2]を表示）．図より，提案手法が生のブラウザと比較して遜色ない再現性を実現することがわかる．

3.1 オリジン

我々は，危険箇所の判別に，JavaScript の同一オリジンポリシーの概念を準用した．具体的には，JavaScript 以外のリソース（例：画像・動画）に対しても，自オリジンであれば信頼でき，他オリジンであれば危険箇所と判別する．

3.2 HTML タグ

我々は，危険箇所の判別に，オリジンに加えて HTML タグ（以下，タグ）も使用した．これは，オリジンのみで判別すると，危険箇所とみなされる範囲が広くなりすぎてしまうためである．具体的には，他オリジンへの遷移を行うタグ（例：<a>）と他オリジンからリソースを読み込むタグ（例：<img>）とを危険箇所と判別する．ただし，危険箇所と判別するタグの種類は，容易に変更可能である．

4. 設計

提案手法の概要を図 2 に示す．我々は，先行研究（2 節）を踏襲しつつ，中継サーバ（4.1 節）と画像化サーバ（4.2 節）とを再設計した．具体的には，部分画像化を可能にするための改良を先行研究に施した．

図 1 生ブラウザと提案手法との実行画面の比較
Figure 1 Comparison between raw browser and proposed method.

表 1　各 Web サイトのトップページを用いた生ブラウザと全体画像化と部分画像化との比較結果
Table 1 Comparison among raw browser, virtual browser, and partial virtual browser by using top pages of websites.

	生ブラウザ（Firefox）		全体画像化（SlimerJS）		部分画像化（SlimerJS）	
Web サイト	転送量（KB）	転送時間（ms）	転送量（KB）	転送時間（ms）	転送量（KB）	転送時間（ms）
Yahoo! Japan	1,080	3,420	1,641	9,810	1,588	9,010
Fc2.com	333	960	358	6,670	231	2,130
niconico	909	22,720	1,265	35,310	1,102	28,860

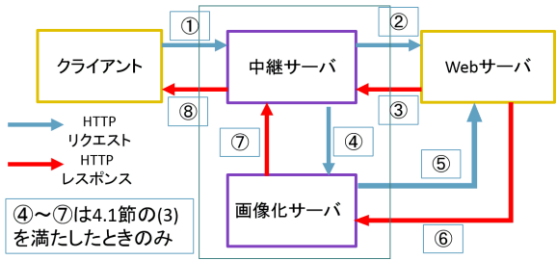

図 2　提案手法の動作概要
Figure 2 Overview of proposed method.

4.1　中継サーバ

中継サーバはフォワードプロキシとして動作し，以下の処理を行う：（1）閲覧者のブラウザ（以下，クライアント）から HTTP 要求を受け取り（図 2 の①），（2）閲覧対象の Web サーバ（以下，Web サーバ）へ HTTP 要求を転送し（同②），（3）Web サーバからの HTTP 応答（同③）の種別が HTML ファイル（以下，HTML）であれば（Content-Type=text/html ならば），クライアントからの HTTP 要求を画像化サーバに転送し（同④），（4）クライアントに画像化サーバからの HTTP 応答を返す（同⑦，⑧）．また，（3）の条件に該当しない HTTP 応答（同③）は，そのままクライアントに転送される（同⑧）．

4.2　画像化サーバ

画像化サーバは画像化のために以下の処理を行う：（1）中継サーバから HTTP 要求を受け取り（図 2 の④），（2）Web サーバへ HTTP 要求を出し（同⑤），（3）Web サーバからの HTTP 応答（同⑥）として得た HTML 中の危険箇所を判別し，（4）危険箇所のみを外見が等価な画像に置換し，（5）部分画像化した HTML を中継サーバに HTTP 応答として返す（同⑦）．部分画像化は，Base64 化した PNG 画像を `<canvas>` 要素の配下に埋め込むことで実現した．

5. 実装

提案手法のプロトタイプの実装には，Node.js[3] と SlimerJS[4] とを使用した．Node.js は，サーバサイドの JavaScript フレームワークで，中継サーバの実装に使用した．SlimerJS は，GUI を必要とせず動作する JavaScript 用のブラウザで，画像化サーバの実装に使用した．SlimerJS は Firefox と同じレンダリングエンジンを用いるため，生のブラウザ（Firefox）との比較評価（6 節）のために採用した．

6. 評価

我々は，部分仮想化の有効性を評価するために，生のブラウザと全体画像化と部分画像化とを，応答性と再現性とについて比較した．応答性の評価には，尺度として転送量と転送時間とを用いた（図 2 の⑧）．再現性の評価は目視で行った．評価対象には，日本の Web サイトのトラフィック量の上位 10 サイト[5]から，HTTP のみで通信できる 3 サイトを使用した．

応答性の評価結果を表 1 に示す．まず，全体画像化と比べると，部分画像化は転送量と転送時間とを削減できている．これは，提案手法によって画像化の範囲が縮小したために，画像化後の容量と画像化にかかる時間とが減少したためである．次に，生のブラウザと比べると，部分仮想化によって転送量と転送時間とがおおむね増加している．転送量は，画像化した対象によって変化するが，画像化した箇所の数とともに増加する傾向にある．転送時間は，画像化処理（例：HTTP 通信，画像変換）によって増加する．

再現性を定量的に評価することは難しいが，目視による定性的評価の結果，生のブラウザと遜色ない画面表示を提案手法が実現できることを確認した（図 1）．また，表 1 の評価対象サイトにおいて，危険箇所と判別しなかった部分の動的コンテンツが正しく動作することを確認した．

我々は，これらの評価結果から，先行研究の仮想ブラウザのユーザビリティの向上に提案手法が有用であるという結論を得た．

7. おわりに

本稿では，先行研究の仮想ブラウザのユーザビリティを向上させるために，オリジンとタグとを用いて危険箇所のみを判別して画像化する部分仮想化手法を提案した．我々は，評価の結果から，提案手法が仮想ブラウザのユーザビリティの向上に有用であるという結論を得た．

今後の課題としては，提案手法のオーバーヘッドのさらなる削減による応答性の向上や，危険箇所の判別情報をオリジンやタグ以外に増やしての再現性の向上を考えている．

謝辞

本研究は JSPS 科研費 26730041 の助成を受けたものです．

参考文献

[1] 早川智一，疋田輝雄：HTML5 を用いた仮想 Web ブラウザの提案と評価，情報処理学会論文誌，第 57 巻第 2 号，pp.573-582, 2016.
[2] 情報処理学会第 79 回全国大会, http://www.ipsj.or.jp/event/taikai/79/.
[3]Node.js, https://nodejs.org/.
[4]SlimerJS, http://slimerjs.org/.
[5] Top Sites in Japan, http://www.alexa.com/topsites/countries/JP.