

ホームページ改ざん検知システムにおける 更新作業効率化のための一手法

富永 浩之[†] 高屋敷 光一[†] 猪股 俊光[†] 曾我 正和^{††}

[†]岩手県立大学大学院ソフトウェア情報学研究科 ^{††}岩手県立大学地域連携研究センター

1 はじめに

ホームページに対する改ざん攻撃に対処すべく、筆者らはデジタル署名とパトロールを基にしたホームページ改ざん検知システム iP@TROL[1] の開発を行っている。従来の iP@TROL システムでは、ホームページの作成者がサーバ上のコンテンツの更新を行う際、更新箇所のサイズに関わらず、ホームページを構成する全コンテンツを対象として、デジタル署名の付与および FTP サーバへのアップロードを常に行わなければならない、更新作業に多くの時間を要する点が問題となっていた。

そこで本研究では、更新作業の効率化を目指し、必要最小限のコンテンツのみを更新対象とするための手法を考案し、iP@TROL への実装を試みた。

2 従来のコンテンツ更新

コンテンツの更新処理は、次に示す手順のように二つに大別される。

Step-1 コンテンツ全体を対象として 1 個の署名データを生成する

Step-2 一旦 FTP サーバ上のコンテンツ全体を削除したのち、署名データと全コンテンツを FTP サーバにアップロードする

ここで、全コンテンツとは、ユーザのホームページ全体を構成している全ファイルを指す。通常、更新処理全体に占める Step-2. アップロードの割合は約 9 割であり、この時間は全コンテンツのサイズに応じて増大する。また、ユーザの全コンテンツに対して 1 個の署名データを作成するため、たとえ更新が必要なファイルが 1 個だけであっても、全コンテンツに対する署名データの生成とアップロードを行う必要がある。

3 差分更新方式

本研究では、更新作業時間の短縮化を目的として、必要なファイルのみをアップロードする「差分更新方

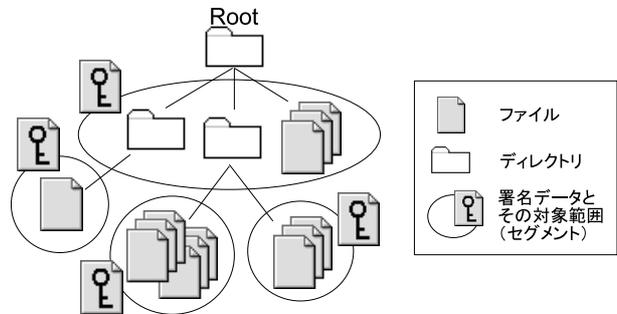


図 1: コンテンツと署名の対応

式」を考案した。その概要を以下に示す。

セグメント 署名データは図 1 のように 1 個または複数個のファイル（ディレクトリも含む）ごとに作成される。このときのファイルの集まりをセグメントと呼ぶ。

セグメント定義ファイル 署名データとコンテンツの対応、コンテンツの更新日時などが記録される図 2 のようなファイル。このファイルは更新作業における更新対象ファイルを抽出するために利用される。各行をレコードと呼び、レコードにはファイルの名前やサイズ、タイムスタンプが記述される。

セッション 1 個のセグメント定義で表されるコンテンツ管理対象の単位をセッションという。セッションは図 3 のように、コンテンツ・セグメント・サーバに関する設定を保持する。これらの設定内容とセグメント定義の対が 1 個のセッションとして扱われ、これによって複数個のホームページそれぞれの更新状況が管理される。

更新リスト・削除リスト・セグメントリスト 更新リストと削除リストには、それぞれ、更新対象と削除対象のファイルの情報が格納される。セグメントリストには、更新リストか削除リストに含まれるファイルを含むセグメントの情報が格納される。

提案方式の更新手順

Step-0 セグメント定義ファイルが存在しない場合（新規アップロード時など）、更新対象ディレクトリについてのセグメント定義を作成する。これに含まれるすべてのレコードを更新リストおよびセグメントリストに追加する。

A Method for Efficient Updating Contents in the Change Detection System for Web Pages

[†] Hiroyuki TOMINAGA, Kouichi TAKAYASHIKI, Toshimitsu INOMATA

^{††} Masakazu SOGA

Graduated School of Software and Information Science, Iwate Prefectural University ([†])

Iwate Prefectural University Regional Cooperative Research Center (^{††})

```

00000000, !:/<0>
00000001, /content
00000002, /graphs
00000003, 1.txt|177|-145710592|
00000004, 2.txt|402|-1013976576|
00000005, conts.htm|3128|+748324096|

```

図 2: セグメント定義の例

```

[Contents]
UserName=TESTUSER
LocalRoot=D:/Webs/WebTest
SegmentSize=10000
SegmentCount=50
[FTP]
Address=192.168.1.2
UserName=Admin
HostRoot=public_html
Passive=0
[Check]
Value=35112

```

図 3: セッション設定の例

セグメント定義ファイルが存在する場合、その内容をもとに現在のファイル構成との差分（更新すべきファイル群）を抽出し、更新リスト・削除リスト・セグメントリストに追加する。このとき一切の差分が検出されなかった場合、更新処理を終了する。

- Step-1 セグメントリストに記述されているセグメントごとに署名データの生成を行う。
- Step-2 削除リストに含まれるファイルについてFTPサーバに削除要求を出したのち、更新リストに含まれるファイルをFTPサーバへアップロードし、同様にセグメントリストに含まれるセグメントに対応した署名データもアップロードする。

4 評価

自動的にセグメント定義・署名データの生成・アップロードまでを行う更新作業用プログラムを作成し、ユーザマシン（CPU2GHz・RAM768MB）、FTPサーバ（CPU333MHz・RAM256MB）、LAN（100Mbps）の環境、全コンテンツを64kバイト×1024個のファイルとし、ファイル50個ごとにセグメントを生成したのち、提案した差分更新方式の性能を評価した。その結果を図4に示す。

図4より、提案方式における全コンテンツ更新である“新-全体”では3節のStep-2にかかる時間が97%を占めた。1個のファイルのみを更新した場合、提案方式である“新-1個”は、従来方式にあたる“旧-全体”の約1/45の時間で更新処理を完了した。なお、図4において“新-全体”が“旧-全体”よりも更新処

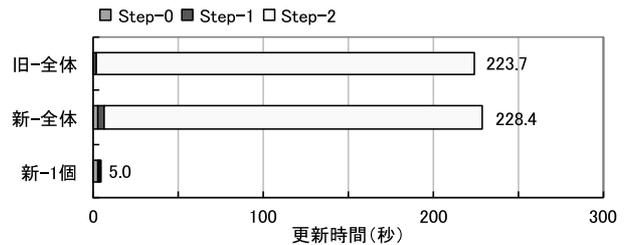


図 4: 更新処理時間の比較

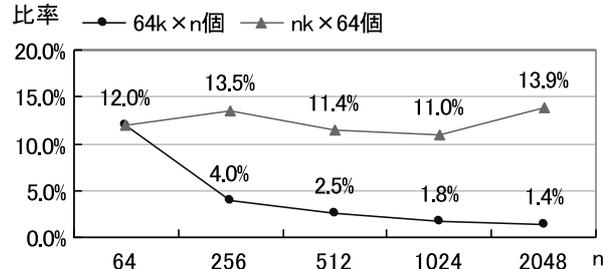


図 5: 全コンテンツ更新時間に対する最小更新時間の比率

理時間が長いのは、Step-0が加わったことと、Step-1で生成すべき署名データの個数が増加したことに起因する。

図5は、提案方式において、コンテンツのファイルサイズを64kバイトとしファイル数 n を変動させた場合と、ファイル数を64個としファイルサイズ n を変動させた場合の、全コンテンツ更新時間に対するファイル1個の更新時間（最小更新時間）の比率を示したグラフである。ファイルサイズを増加させた場合、比率は10~15%の間で時間短縮の効果がほぼ一定である。これに対し、ファイル数を増加させた場合の方が、差分更新による時間短縮の効果が著しい。このことから、差分更新方式は特にファイル数が多いホームページに対して有効といえる。

5 おわりに

本研究では、ホームページの作成者がコンテンツを更新する際の負担の軽減を図るために差分更新方式を提案し、所期の効果が得られることを確認した。今後の課題として、サーバにおけるパトロール処理の性能評価、複数ユーザによる同一ホームページ管理のケースへの対応、さまざまなファイル構成や更新頻度に応じた適切なセグメントの構成法の検討があげられる。

参考文献

- [1] 猪股俊光, 板垣晋, 曾我正和, 西垣正勝: “デジタル署名とパトロールを用いた電子情報改ざん検知方式とWWWへの応用”, 情報処理学会論文誌, Vol.44 No.8, pp.2072-2084(2003)。