L-015

DNSによるインシデント拡散検知

Incident diffusion detection by DNS

紅谷 知輝† 鈴木 常彦†
Tomoki Beniya Tsunehiko Suzuki

1. はじめに

近年，インターネットの普及に伴ってワーム型ウイルス（以降，ウイルスと呼ぶ）による大規模感染が起きるようになり，その被害も深刻なものとなっている．主な理由としては，拡散にネットワーク（主に電子メール）を利用したり，OSやソフトウェアのセキュリティホールを狙うようになったことが挙げられる．

また，現在主流のウイルス検知方式である「パターンマッチング」は，常にパターンファイルを最新のものにしなければならないが，最近のウイルスは，電子メールを利用して爆発的なスピードで広がるため，パターンファイルがユーザに届く前に被害が拡大してしまっている．

そこで，本論文では，ウイルスの振る舞いを利用し，ウイルス対策ソフトを導入することなくウイルス拡散を防止し，第三者に利用されることを防止するシステムを提案する．

2. 通信の相違点

ウイルスに感染した端末には特徴的な通信が発生する．

ホームページの閲覧や電子メールの送受信のような一般的な利用なら，端末からDNSキャッシュサーバに対してAレコードの問い合わせしか発生しない．これは，ウイルスに感染していても，していなくても共通に行われる．しかし，ウイルスに感染した端末は，ウイルスが持つ独自のSMTPエンジンを利用し，メーラーやメールサーバを利用せず直接送信相手のメールサーバに電子メールを送りつけることで感染拡大を試みるため，通常では有り得ない「MXレコード」の問い合わせが発生する．

3. システムの設計

3.1 感染端末の特定

感染端末の特定を行うには，2で述べたウイルスに感染した際に発生する「MXレコードの問い合わせ」を捕捉する必要がある．MXレコードの問い合わせは，DNSキャッシュサーバに対して行われるため，DNSキャッシュサーバ内にDNSの問い合わせログを監視するプログラムを作成し，感染端末の特定を行う．端末の特定には，ネットワーク上で重複しないIPアドレスを利用する．ログの中からパターンマッチを用いて「MXレコードの問い合わせ」を取り出し，さらにそこから問い合わせ元のIPアドレスを特定し，これを「感染端末」とする．

3.2 感染端末の通信を制限

感染端末の特定が完了したら，まず「ネットワークからの切り離し」を行う．これには，Firewallのパケットフィルタリング機能を用いる．3.1で特定した感染端末のIPアドレスをFirewallへ転送し，Firewall内に作成したプログラムを用いて設定変更を行う．この設定変更は，後述の感染端末利用者への通知のことも考慮に入れて「感染端末からDNSキャッシュサーバの53番ポートと外部webサーバの80番ポートのみアクセス可能」，「外部から感染端末へのアクセスは不可能」という環境を作るこの設定によって，外部から感染端末へアクセスすることはもちろん，感染の拡大を阻止する．

3.3 利用者への通知

利用者への通知は，特殊な操作がなく，分かり易く伝える必要があるため，webブラウザを用いることにした．webブラウザは，文字だけでなく画像等の表示もできるため，対処法や今後の対策法などをより分かり易く伝えることが可能となる．

これを実現するためにローカルwebサーバを用意し，そこへ誘導することで実現する．3.2によって外部webサーバへのアクセスが可能となり，それに加えてFirewallのNAT機能を用い，感染端末から外部webサーバ宛のパケットを強制的にローカルwebサーバ宛に変更する．こうすることによって，どこのホームページにアクセスしても，ローカルwebサーバにアクセスすることになる．

4. システムの実装

4.1 システム環境

本システムは，DNSキャッシュサーバ，Firewall，webサーバの３つを組み合わせることで実現する．特に，DNSキャッシュサーバとFirewallの連携が重要である．これら3つには，Linux系，BSD系のOSを用いることにした．

4.2 システムの構築

DNSキャッシュサーバには，djbdnsのdnscacheを利用して実現し，このログファイルを監視するプログラムを作成する．また，Firewallへ感染端末のIPアドレスを送るためにSyslog-ngのログ転送機能を用いる．

Firewallには，FreeBSDが標準でサポートしているPacket Filter（PF）を用いる．また，DNSキャッシュサーバから送られてくる感染端末のIPアドレスを受け取るために，ここでもSyslog-ngを利用し，Firewallの設定変更を自動化するためのプログラムも作成する．PFが実行中に変更できるものは，Tablesというアドレスリストのみなので，前もって全てのフィルタリングルールやNATルールを作成し，感染端末制限用のルールには，変更可能なTablesを設定し，それを操作する．

webサーバには，現在最も利用されているApacheを用いる．本来ならば，公開用フォルダにファイルを置くだ

†中京大学

けでよいのだが，NAT によって曲げられたアクセスを受けるため，大半が存在しないファイルへのアクセスになると考えられる．そこで，mod_rewrite モジュールを組み込み，存在しないページへアクセスでもリダイレクトを行い存在するページを表示させる．

4.3 システムの構成

本システムの構成は図 1 のようになる．Firewall で新たにサブネットを作成し，その中の端末をウイルス検知の対象にする．また，メールサーバは，既存のものを流用するため構築していないが，便宜上，記しておく．

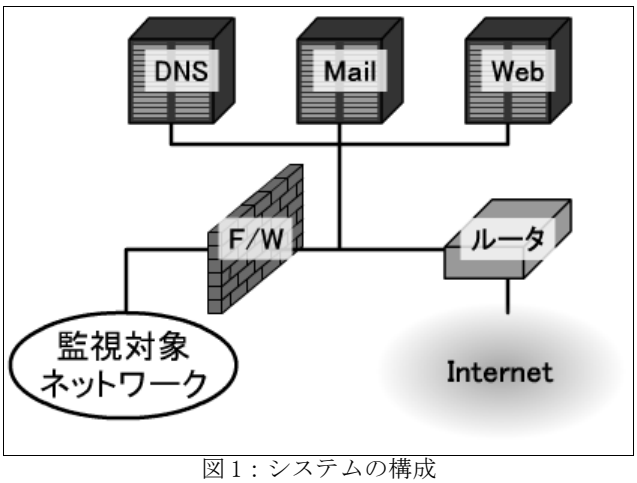

図 1：システムの構成

5. 評価

5.1 動作確認

W32/Netsky@mm，W32/Beagle@mm，W32/Klez@mm の現存する３つのワーム型ウイルスを使ってシステムが正しく動作するか確認を行った．確認環境には，WindowsXP SP2 上で Virtual PC 2004 を用い，その仮想空間内で比較的ウイルスに感染する可能性が高い Windows2000 SP4 を用いた．

```
01-15 12:36:53 sent 23
01-15 12:36:53 query 25 192.168.150.7:1037:10653 mx example.com.
01-15 12:36:53 cached mx example.com.
01-15 12:36:53 sent 25
01-15 12:36:53 nodata 193.0.0.236 3600  15 example.com.
01-15 12:36:53 stats 25 49249 1 0
01-15 12:36:53 sent 24
01-15 12:36:53 query 26 192.168.150.7:1038:23453 mx example.com.
01-15 12:36:53 cached mx example.com.
01-15 12:36:53 sent 26
01-15 12:37:06 query 27 192.168.150.7:1039:30156 mx example.com.
01-15 12:37:06 cached mx example.com.
01-15 12:37:06 sent 27
01-15 12:37:06 query 28 192.168.150.7:1040:45516 mx example.com.
01-15 12:37:06 cached mx example.com.
```

図 2：DNS キャッシュサーバのログの様子

W32/Netsky@mm，W32/Beagle@mm は，感染と同時に端末内で収集したメールアドレスに対して独自の SMTP エンジンを利用した電子メールの送信と思われる動作を確認した（図 2 の枠内）．そのため，DNS キャッシュサーバでそれらを検知し感染端末の通信を制限することができた．また，その端末の web ブラウザで外部サイトの閲覧を試みると，ローカル web サーバのページが表示され，システムが，正常に機能していることを確認できた．

しかし，W32/Klez@mm は，電子メールの送信方法は先のウイルスと同じだが，送信先のメールサーバのアドレスを予測し，その A レコードの問い合わせを行うため，検知できなかった．

5.2 考察

今回，既知のウイルス 3 種類を用いて動作確認を行ったが，MX レコードの問い合わせのみを監視対象としたため，MX レコードを利用する W32/Netsky@mm，W32/Beagle@mm は，問題なく検知できたが，W32/Klez@mm のような MX レコードを利用しないウイルスの検知はできなかった．一方，感染端末利用者に対する通知は，「web ブラウザでどこかのホームページを閲覧する」という，非常に簡単な操作で自身がウイルスに感染しているかどうかを知ることができるようになり，Firewall による通信制限によって大量メール送信や DoS 攻撃のプラットフォームなど，第三者に悪用される恐れもなくなった．また，一番初めの端末がウイルスに感染してしまうことを防ぐことはできないが，他の端末への感染を防ぐことで被害を最小限に抑えることが可能となった．

5.3 関連研究

ウイルスに関する研究は仮想環境内でウイルスの動きを観測し，検知するものが多く，本論文のようなウイルスの特徴的な動作を実際に扱ったものは数が少ない．[1] では本論文と同様に DNS クエリを監視することで感染端末の特定を試みており，Netsky や Sobig の感染を確認できているが，感染端末の特定に重点を置いているためか，端末特定後は管理者にメールが送られるだけである．しかし，本論文では感染端末を特定を行うのはもちろん，感染端末に対して Firewall による通信制限を設け，更に利用者への通知も可能にしている．

5.4 今後の課題

MX レコードを利用するソフトウェアや手動による MX レコードの問い合わせにを行った際に起きる誤検知の問題がある．この問題には，MX レコードの問い合わせ頻度を組み合わせることで解消されるものと思われる．また，W32/Klez@mm のようなメールサーバを予測するウイルスに対しては，実際に問い合わせられた A レコードとそのドメインの MX レコードを照らし合わせることで回避できるものと考えられるが，今回はどちらも実装にまで至っていない．さらに，ネットワークを利用するため，その混雑状況によっては少なからず遅延が発生してしまう可能性がある．

謝 辞

本研究を行うにあたり，お忙しい中，時間を割いて多くのご指導，助言をいただいた鈴木常彦准教授ならびに山口榮作先生に心から感謝する．

参考文献

[1] 武藏泰雄，Kai Rannenberg：DNS クエリアクセス監視による大量メール送信型ワーム感染端末の検知
http://www.cc.kumamoto-u.ac.jp/~musashi/musashicsec27.pdf