Vol.2014-DPS-158 No.34
Vol.2014-CSEC-64 No.34
2014/3/7

# マルウェア通信解析システムの検討

大越冬彦[†1]　　桜井鐘治[†1]

マルウェアの暗号通信を復号して通信内容を解析するためには、復号方法であるマルウェアの解析を行って暗号鍵を抽出する方法と復号関数を特定することが必要である。このマルウェア解析には解析そのものの作業や関連するツールの作成など多数の工数がかかり、迅速な解析の足かせとなっていた。

本システムは既知のマルウェア解析結果から得た復号方法とマルウェアを識別するマルウェアシグニチャを用いて、マルウェアが行った暗号化通信の復号を行う。復号方法とマルウェアシグニチャをマルウェア情報データベースに保存し、マルウェア通信解析ツールから呼び出すことにより複数のマルウェアに対応することが可能になり、マルウェア解析の効率を向上させる。

In order to decode the encryption communication of malware and to analyze the contents of communication, it is required to specify the method of analyzing the malware which is the decoding method and extracting an encryption key, and the decoding function uses. Malware analysis took time and effort, such as work of the analysis itself, and creation of analysis tools, and quick analysis was difficult. This system decodes the encrypted communication which malware performed using decoding method and malware signature which were obtained from the result of having analyzed known malware are used. Decoding methods and malware signatures are saved at a malware information database, and it becomes possible to correspond to multiple malware by calling from malware communication analysis tools, and raises the efficiency of malware analysis.

# Examination of a malware communication analysis system

FUYUHIKO OKOSHI[†1]　　SHOJI SAKURAI[†1]

## 1. はじめに

昨今、マルウェアを用いた標的型攻撃による情報漏洩事件が頻発している。マルウェアによる感染が発覚したときには、そのマルウェアが外部に送信したデータから漏洩した情報を特定して、被害を明確にする必要がある。

従来、マルウェアによる感染が検出された場合、感染した端末の HDD やサーバログからマルウェアがアクセスしたファイルを特定する。また、プロキシなどネットワーク機器のログからマルウェアがアクセスした通信先を特定などの手法で被害範囲を特定してきた。しかしながら、これらの手法では、漏洩した可能性があるデータの候補は絞れるが、実際に漏えいしたかを確定することは困難であった。[1]

これに対して、インターネットと組織内ネットワーク上の情報処理端末が行なった通信をネットワークフォレンジック装置により記録しておき、マルウェアの感染した端末の通信記録を抜き出して解析する手法が有効である。しかし最近のマルウェアのほとんどは､アンチウィルスソフトや､侵入検知システムなどに対抗するために、通信を暗号化しており、単に通信を記録しておくだけでは、漏洩した情報を特定することは困難である。[2]

†1 三菱電機株式会社 情報技術総合研究所
Information Technology R & D Center, Mitsubishi Electric Corporation

このためマルウェアが行った暗号通信を復号して通信内容を明らかにする必要がある。

## 2. マルウェア解析の課題

マルウェアは暗号通信を復号するための復号関数を内蔵している場合がある。また通信の復号のための暗号鍵は、マルウェア内部に隠されている場合や、通信に含まれるデータから生成される場合がある。これらの場合、マルウェアを解析し、暗号鍵を抽出する方法と復号関数が利用しているアルゴリズムを特定すれば、マルウェアが行った暗号通信を復号することが可能である。

しかしながらこのマルウェア解析には技術者による解析作業が必要となる。また鍵抽出方法と復号関数が明らかに出来たとしても、それらを用いて暗号化通信を復号するツールの作成が必要であり、さらに工数がかかる。

## 3. 解決策

これらの課題を解決するために、これまで解析ごとに異なっていた、マルウェア解析における暗号鍵の抽出方法と復号関数の形式を標準的な形式として、それらをデータベース化して、マルウェアに関する情報とともに保存することにより、マルウェア解析の作業を効率化するシステムを作成する。図 1 に本システムの概要を示す。

Vol.2014-DPS-158 No.34
Vol.2014-CSEC-64 No.34
2014/3/7

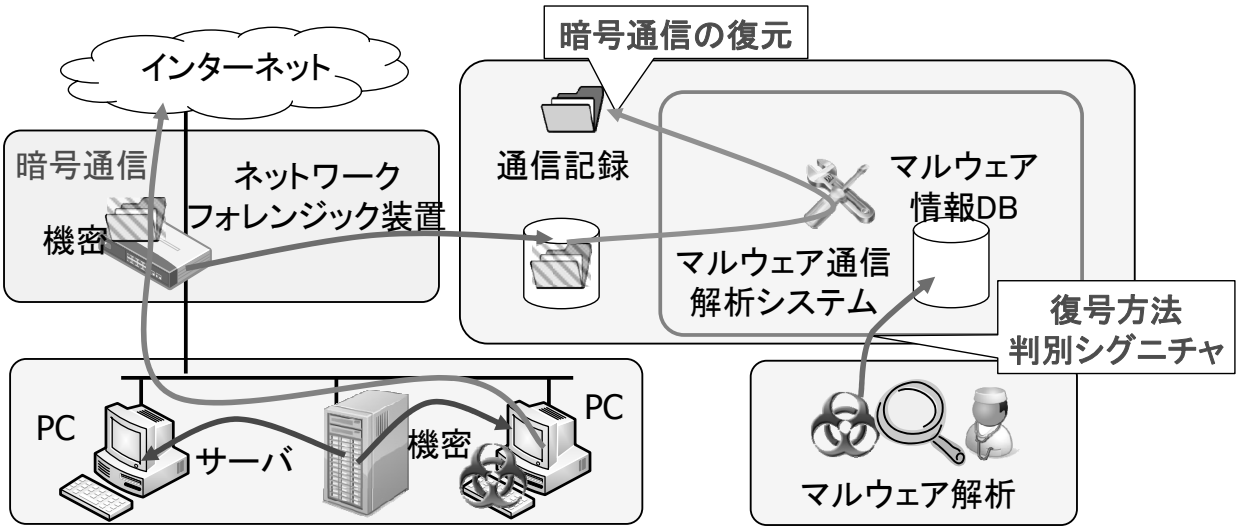


図 1 システムの概要

## 4. 構成

今回作成するマルウェア通信解析システムは

- 複数のマルウェアを解析した結果から得たマルウェアの通信の特徴
- マルウェアが使用している暗号鍵の抽出方法
- マルウェアから抽出した復号関数

を記録したデータベースを通信記録と照合することにより、マルウェアの種類を特定しそのマルウェアに適合した暗号鍵の抽出方法と復号関数を用いて、暗号化された通信を復号することにより、マルウェアが行った通信内容を効率的に特定する。

本システムの構成を図 2 および表 1 に示す。

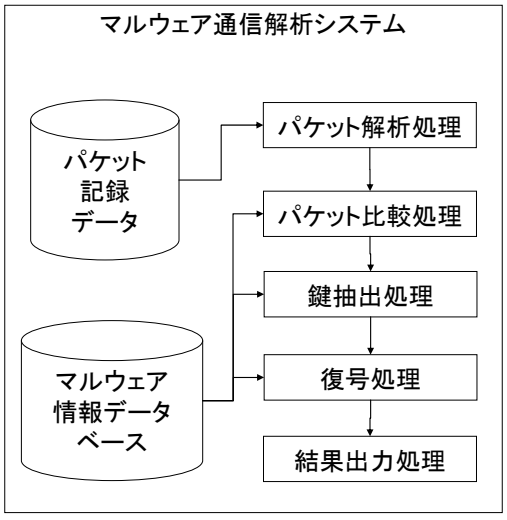


図 2 システムの構成

表 1 構成

| 名称 | 機能 |
|---|---|
| パケット解析処理 | パケット記録データから記録された順にパケットを取り出し、パケットを IP 情報、トラスポート情報、ヘッダー情報、ペイロード情報に分割したパケット解析データを生成する。 |
| パケット比較処理 | パケット解析データとマルウェア情報データベースに記録されたシグネチャとを比較し、一致するかを判定する。 |
| 鍵抽出処理 | 一致したシグネチャに対応する鍵抽出情報を読み取り、その記述に従ってパケット解析データから暗号鍵を抽出する。 |
| 復号処理 | 一致したマルウェアシグネチャに対応する復号関数情報を用いて、パケット解析データのペイロード情報に記録された暗号データを暗号鍵で復号して復号データを得る |
| 結果出力処理 | 復号データを出力する。 |
| パケット記録データ | インターネットとの通信パケットを記録したデータであり、取得方法は問わないが、過去に取得したデータであっても、リアルタイムで取得したデータも良い。 |
| マルウェア記録データベース | マルウェアが行う通信の特徴を表したシグネチャと鍵抽出情報と復号関数情報を保持する。 |

Vol.2014-DPS-158 No.34
Vol.2014-CSEC-64 No.34
2014/3/7

### 4.1 鍵抽出処理

鍵抽出処理の構成を図 3 に示す。鍵抽出処理は鍵抽出情報解釈処理と抽出実行処理から構成される。鍵抽出情報解釈処理は鍵抽出情報を解釈する。抽出実行処理はパケット解析データから鍵抽出を行う。

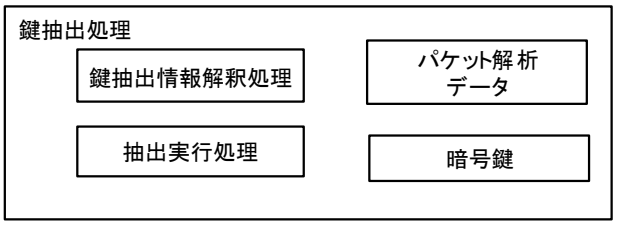


図 3 鍵抽出処理の構成

### 4.2 復号処理

復号処理の構成を図 4 に示す。復号処理は実行制御処理、コードメモリ、スタックメモリ、バッファメモリから構成される。実行制御処理は実行制御情報に基づき、復号関数の実行を管理する。コードメモリはマルウェア情報データベースから読み出したコード情報がロードされる。スタックメモリは復号関数に引数を与えるためのものである。バッファメモリは復号関数との暗号データと復号したデータと暗号鍵情報を交換する。実行制御情報の設定例を表 2 に示す。

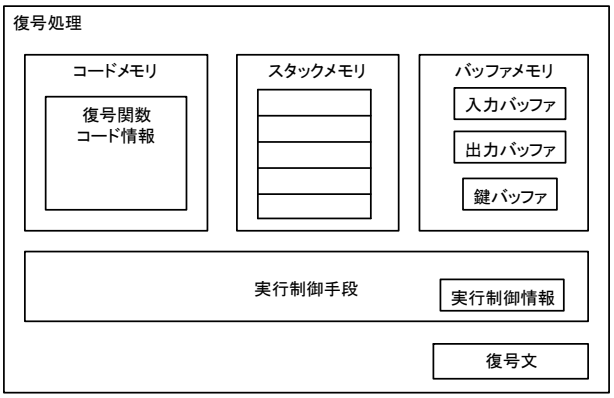


図 4 復号処理の構成

### 4.3 パケット解析データ

パケット解析データの構成例を表 3 に示す。IP 情報、トランスポート情報、ヘッダ情報、ペイロード情報で構成されている。

### 4.4 マルウェア情報データベース

マルウェア情報データベースの構成を表 4 に示す。また設定例を表 5 に示す。

表 2 実行制御情報

| 設定項目 | 設定内容 |
|---|---|
| 入力バッファポインタ | STACK(2) |
| 入力バッファサイズ | STACK(3) |
| 出力バッファポインタ | STACK(4) |
| 出力バッファサイズ | STACK(5) |
| 暗号鍵バッファポインタ | STACK(1) |
| 暗号鍵長 | NULL |
| 返り値 | AX |

表 3 パケット解析データ

| 分類 | 書式 | 値 | 備考 |
|---|---|---|---|
| IP 情報 | SRC | 10.74.5.112 | IP プロトコルに関する情報 |
| | DEST | 168.192.100.1 | |
| トランスポート情報 | PORT | 80 | TCP/UDP プロトコルに関する情報 |
| | FLAG | ACK PSH | |
| ヘッダ情報 | HTTP.URL | STACK(1) | TCP/UDP プロトコルで運ばれる上位プロトコルのヘッダー情報 |
| | HTTP.VBNET | 2334 | |
| ペイロード情報 | BODY | (バイナリ値) | TCP/UDP プロトコルで運ばれる上位プロトコルのペイロードの情報 |

表 4 マルウェア情報データベース

| 項目 | | 内容 |
|---|---|---|
| シグネチャ | | マルウェアが行う通信の特徴を表す |
| 鍵抽出情報 | | 通信パケットから暗号鍵を抽出する方法を記述したスクリプト情報 |
| 復号関数情報 | コード情報 | 復号関数を実装した命令コード |
| | 実行制御情報 | 復号関数を呼び出すための引数の順序および返り値を示すＣＰＵレジスタ情報を記述 |

Vol.2014-DPS-158 No.34
Vol.2014-CSEC-64 No.34
2014/3/7

表 5 マルウェア情報データベース例

| シグネチャ | 鍵抽出情報 | 復号関数情報 |
|---|---|---|
| HTTP.HEADER(Request URI)==”//www.malware.com” | $KEY=”0xDFC14B23”; | コード情報<br>実行制御情報 |
| IP.SRCADDR==”10.74.5.112” | $KEY=HTTP.Header(SECRETKEY); | コード情報<br>実行制御情報 |
| HTTP.HEADER(VBNET) ~= “[0-9]*” | $work=mid(HTTP.PAYLOAD,16,4);<br>$KEY=xor($work,”OxFDFDFDFD”); | コード情報<br>実行制御情報 |

## 5. 処理内容

本システムの処理フローを図 5 に示す。

### 5.1 パケット解析処理

パケット解析処理は、パケット記録データから記録された順にパケットを取り出し、パケット分析処理は、取り出したパケットを IP 情報、トランスポート情報 、ヘッダー情報、ペイロード情報として分割したパケット解析データを生成する。

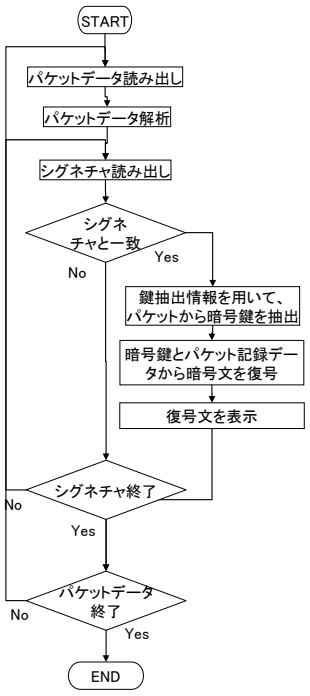


図 5 システム処理フロー

### 5.2 パケット比較処理

パケット比較処理はマルウェア情報データベースからシグネチャ情報を読み出し、パケット解析データとシグネチャ情報のパターンマッチングを行い一致するかを検証する。シグネチャ情報は各マルウェアに関する IP 情報、トラスポート情報、ヘッダ情報のいずれかを指定する情報と、IP 情報、トラスポート情報、ヘッダ情報のいずれかの値を表す正規表現と、比較方法を示す論理演算子から構成されている。

- HTTP.HEADER(Request URI)==”//www.malware.com” と記述されていた場合、対象としてヘッダ情報を選択しており、プロトコルが HTTP であり、Request URL が//www.malware.com と同一であった場合に一致する。例えば HTTP.HEADER(VBNET) ~= “[0-9]*”と記述されていた場合、対象としてヘッダ情報を選択しており、プロトコルが HTTP であり、ヘッダが VBNET というヘッダが存在し、ヘッダの値として数値が記述されている場合に一致する。
- IP.SRCADDR==”10.74.5.112”と記述されていた場合、対象として IP 情報を選択しており、発信元の IP アドレスが 10.74.5.112 であった場合一致する。
- 記述には論理演算子として論理積 AND、論理輪 OR、否定 NOT を使用することができ、複数の記述を組み合わせることができる。
- パケット比較処理にてパケット解析データとシグネチャ情報が一致した場合には、パケット解析データに関わるパケットはマルウェアが行った通信であるとみなすことができる。一致しなかった場合には、シグネチャ情報が終了していない場合には次のシグネチャ情報を読み出す処理を繰り返す。

### 5.3 鍵抽出処理

鍵抽出処理はマルウェア情報データベースから鍵抽出情報を読み出す。マルウェアが持つ暗号鍵は、

- マルウェア本体に埋め込まれている固定鍵である場合
- 通信パケットの暗号文に鍵もしくは鍵に関連する情報が含まれている場合

がある。鍵抽出情報は通信パケットから暗号鍵を抽出する方法を記述したスクリプト情報であり、パケット中から任意の位置にある任意の長さのバイト列を選択し、鍵を抽出するために必要な演算を行う。鍵抽出情報の記述例は以下の通り

- 鍵が固定鍵で 16 進数の DFC14B23 であった場合には $KEY=”0xDFC14B23”;
- 鍵が通信パケットの中に含まれている場合で HTTP プロトコルの SECRETKEY ヘッダに鍵が存在する場合、$KEY=HTTP.Header(SECRETKEY);
- 鍵が通信パケットの中に含まれている場合で通信パケ

ットのペイロード情報の先頭から 16 バイト目から 4 バイトを抜き出した値と 16 進数 FDFDFDFD と排他的論理和を取った値が鍵になる場合

$WORK=mid(HTTP.PAYLOAD,16,4);

$KEY=xor($WORK,”OxFDFDFDFD”);

鍵抽出情報解釈処理は鍵抽出情報を読み取り、その先頭部分から記述内容に応じて、抽出実行処理を通じてパケット解析データに対して操作を行い、バイト位置の特定、バイト列の抜き出し、抜き出したバイト列の演算を操作指示に応じて実行することにより、暗号鍵が抽出される。

### 5.4 復号処理

復号処理の処理フローを図 6 に示す。

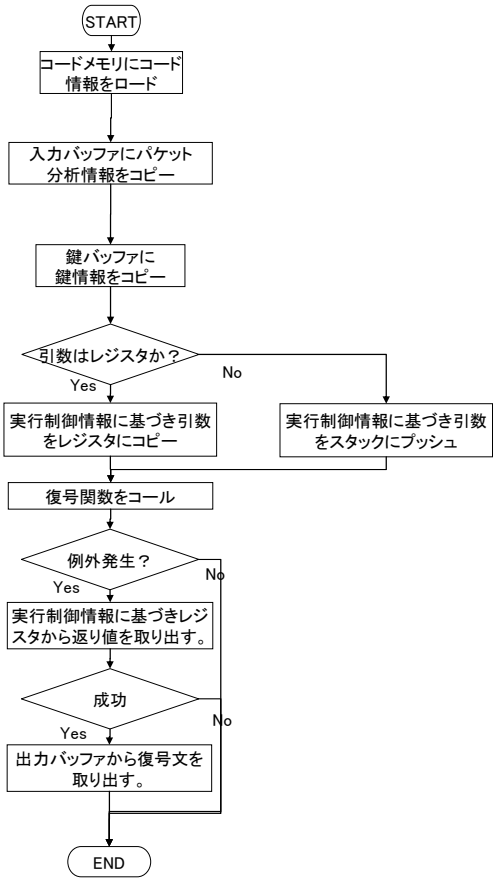


図 6 復号処理フロー

- 実行制御処理はマルウェア情報データベースからコード情報を読み出しコードメモリにロードする。実行制御処理はパケット解析データのペイロード情報を入力バッファにコピーするとともに、暗号鍵を鍵バッファにコピーする。
- 復号関数はマルウェアごとにスタックにプッシュする引数順序が異なり、また特定の CPU レジスタを介して引数が渡される場合もあるため、実行制御情報の設定内容に基づいて、入力バッファの先頭アドレス、出力バッファの先頭アドレス、鍵バッファの先頭アドレスをスタックにプッシュするかレジスタに渡すかを決定する。
- 設定項目が入力バッファポインタであった場合に対応する設定内容が STACK(2)と記述してあれば、入力バッファポインタを 2 番目にスタックにプッシュしてから復号関数を呼び出す。
- 設定項目が出力バッファポインタであった場合に対応する設定内容が DI と記述してあれば、DI レジスタに出力バッファポインタをコピーしてから復号関数を呼び出す。
- 実行制御処理はコードメモリにロードされたコード情報をコールする。入力バッファの暗号データを復号した結果が出力バッファに出力される。
- 実行制御処理は関数コールが実行時例外なしに終了したことを確認し、設定項目に返り値が記述された CPU レジスタ値を読み出し、返り値を判定する。判定の結果、成功していた場合には出力バッファの内容を復号文とする。

### 5.5 結果出力処理

結果出力処理はパケット解析データと復号文の表示を行うとともに、まだパケット記録データが終了していなければ次のパケット記録データの解析を繰り返す。

## 6. 考察

本システムは．複数のマルウェアを解析した結果から得たマルウェアの通信の特徴と、マルウェアごとの暗号鍵の抽出方法と、マルウェアから抽出した復号関数をデータベースに登録しておき、マルウェアの通信の特徴をマルウェアの通信記録と照合してマルウェアの種類を特定する。特定されたマルウェアに適合した暗号鍵の抽出方法と復号関数を用いて、暗号化された通信を復号することにより、

- 既知のマルウェアが行った通信について復号が可能になる。
- 既知マルウェアの亜種のマルウェアについては、復号はできないものの、鍵抽出方法や復号関数が類似している場合があり、比較的対応が可能である。
- 完全に未知のマルウェアの場合は、鍵抽出方法と復号関数を特定する必要が発生する。しかしながらパケットから HTML などのアプリケーション通信部分を抜き出する処理などの共通の部分については再利用が可能であるので、解析に関する全体の作業量は減少する。

本システムにより、マルウェアが行なった通信内容を効率的に特定することが可能になる。

## 7. おわりに

今後は、不審な暗号通信を行うプログラムを自動で解析し

Vol.2014-DPS-158 No.34
Vol.2014-CSEC-64 No.34
2014/3/7

て得られる暗号通信に関する情報から、暗号通信をリアルタイムで復号し、暗号通信がマルウェアによるものかを判定し、攻撃を遮断する手法の検討を実施する。

## 参考文献

1) S.Anson, S.Bunting, R.Johnson, S.Pearson, Mastering Windows Network Forensics and Investigation
2) S.Davidoff , J.Ham ,Network Forensics: Tracking Hackers through Cyberspace , Prentice Hall