2009－UBI－21 (7)
2009／3／4

端末監視によるホームネットワーク異常検知システム

國吉 賢吾† 森井 昌克‡

†神戸大学大学院自然科学研究科
〒657-8501 神戸市灘区六甲台町 1-1
069t220n@stu.kobe-u.ac.jp

‡神戸大学大学院工学研究科
〒657-8501 神戸市灘区六甲台町 1-1
mmorii@kobe-u.ac.jp

あらまし 近年，家電機器のネットワーク化に伴い，より高度な機器状態の監視が必要とされている．家電機器の機能障害やマルウェアによる被害など以前は存在しなかった問題が多く発生する可能性があるため，ホームネットワークにおける機器の状態を監視し，異常が発生した場合は早期検知及び復旧しなければならない．従来，家電機器における位置や設定情報等を利用した異常検知手法は提案されているが，機器内に潜伏する動作異常は検知できない．本稿では異常検知を目的とした監視項目を設け，監視対象となる機器内に出現するプロセス集合を状態判定する異常検知手法を提案する．本システムは特にマルウェア侵入検知を目的とした監視項目を設定した．評価の結果，未知のマルウェアに対する検知見逃しをなくし，対象機器状態を収集することで時間経過に伴い誤検知数を漸近的に減少させることを確認した．

Detecting anomalies by monitoring terminals in Home Network

Kengo Kuniyoshi† Masakatsu Morii‡

†Graduate School of Science and Technology, Kobe University
1-1 Rokkodai, Nada-ku, Kobe-shi 657-8501,Japan
069t220n@stu.kobe-u.ac.jp
‡Graduate School of Engineering, Kobe University
1-1 Rokkodai, Nada-ku, Kobe-shi 657-8501,Japan
mmorii@kobe-u.ac.jp

Abstract Recently the home appliances has been connected in network and worked together mutually. However these devices can receive more failures and malicious attacks by malware. So we must detect the anomaly early and recover it automatically. In this paper, we propose a system which monitors the condition on the each aggregate of the runnnig processes to detect the anomaly. In paticular this system intends the anomaly caused by malware. Our method can detect an unknown one and decrease the false positives as time goes on.

1 まえがき

近年，家電機器のネットワーク化が急速に進み，機器間で相互に連携可能なサービスが提供され始めている．白物家電に適応するECHONETやデジタルコンテンツ共有のDLNA規格等が実際製品に適用され，利便性は拡大している．しかし一方で，家電製品を対象としたマルウェア侵入による被害や家電機器自体に発生する障害など，以前より甚大な問題の生起が予想され，ホームネットワークにおける強固な安全対策が必要とされている．従来，家庭内のセキュリティサービスとして火災検知やガス漏れ検知などが存在していた．これら独立したセンサによる異

常検知は個々としては効果的であるが，ホームネットワークには規格化された多数の家電機器が接続されるため，より汎用的な方式で家電機器を一括管理する異常検知が有効となる．

従来，ホームネットワークにおける異常検知手法として家電機器や周辺環境の温度，位置情報などを利用し，対象機器の危険動作や非効率動作を検知する手法が提案されている [1]．各種センサを用いることで，センサ間の連携把握により異常状態の検知が可能となるが，大量のセンサを設置する必要があるために導入コストが大きい．また，DLNA ネットワークにおいてパケット情報を収集し，ネットワーク設定の異常性を検知する手法がある [2] が，機器の動作異常は検知対象ではない．

本稿では，ホームネットワークにおける異常状態の検知を目的とした監視項目を設定し，監視対象とする機器に出現するプロセス集合の状態判定を行う異常検知方式を提案する．ホームネットワーク内に存在する機器に出現するプロセスの状態は類似し，出現数は限られている．本システムは監視対象に導入するエージェントとインターネットを介して監視対象を統括管理するセンタから成る．エージェントで発生した状態の解析は最終的に重複/類似するプロセス集合における監視項目の状態を管理するセンタで行う．本稿では特にホームネットワーク内へのマルウェア侵入検知を目的とした監視項目を設定する．本システムはセンサ等の特別な装置の設置を不要とし，対象機器における異常動作の検知が可能となる．

2　ホームネットワーク異常状態

ホームネットワークにおいて発生する可能性がある家電機器の異常状態を分類する．家電機器の利用において異常の原因となる状況，設定，動作の点から 4 つの項目を異常として定義する．

- 物理異常

  機器接続間の物理層及びデータリンク層に起こる障害．具体的にはケーブルの断線，機器の電源故障などを原因とする．

- ネットワーク異常

  ネットワーク設定などの IP アドレシングにおいて発生する障害．具体的には IP Duplicate や DHCP サーバによる IP アドレス配布ミスなどがある．

- 非規定動作異常

  メーカが規定する動作からの逸脱やユーザが通常使用する動作と異なる状況など，正常な家電における動作でない状態を指す．機器内における動作不具合や不正プログラムによる非正規動作などが挙げられる．

- 危険動作異常

  家電機器において危険な影響が明示的に及ぼされる状態を指す．家電機器における異常な発熱や他の機器への攻撃，個人情報の流出等がある．

本稿では，特にマルウェアの不正侵入を原因とする非規定動作，危険動作の異常検知を目的とした監視項目を設定し，監視対象で出現するプロセス集合の正常・異常判定を行う．

3　端末監視による異常検知方式

3.1　システム構成

提案システムは監視エージェントとセンタの二者で構成される．図 1 にシステム全体図を示す．監視エージェントは監視対象内に導入し，センタはエージェントとインターネットを介して設置する．エージェントは監視対象で動作するプロセス状態をリアルタイム監視する状態検出部，取得した情報をプロファイルを用いて検知対象情報へと解析する状態解析部，DB を用いて検知対象情報の異常状態を検知する異常検知部から成る．センタは検知対象情報の異常状態を検知する異常検知部のみから成る．エージェントとセンタは共に DB を持ち，各異常検知部における状態判定に使用する．

3.2　検知方式

監視対象で発生するプロセス，パケットを正常データと呼ぶ．これら正常データを多数の監視対象から収集し，センタ DB を構築する．DB は動作するプロセス集合を primary key として，各状態における発信パケット分析情報を持つ．

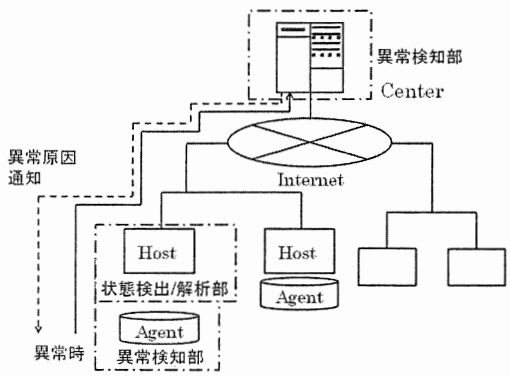

図 1: システム全体図

状態検出部 監視対象で動作するプロセス集合を定期的に取得し，各プロセスの起動状態を監視する．プロセス集合出現中のパケット情報を蓄積保存し，プロセス集合に変化点を検出した場合，状態解析部に通知する．

状態解析部 プロセス集合とパケットから検知対象情報を取得する．プロセス集合から固定，重複，関連プロセス集合を除去し，パケットはDB管理項目毎に解析を行い，エンティティ項目値を取得する．解析終了後，異常検知部へ通知する．

エージェント異常検知部 エージェントDBを用いて状態判定を行う．判定結果が正常時は自身のDBを更新し，異常時はセンタへ検知対象情報と共に解析依頼を送信する．

センタ異常検知部 センタDBを用いて状態判定を行い，判定結果をエージェントへ通知する．正常時はエージェントDBを更新し，異常時は異常検知原因を共に送信してシステム管理者に提示する．

4 構築DB

4.1 エンティティ項目

マルウェア感染時における影響性の取得を目的とし，DBは以下の情報をエンティティ項目として持つ．一般的に発生する影響の監視項目としてProcess Image Name Combination(以下PINC)とOutbound Port，不正侵入・調査活動の監視項目としてMicrosoft-DS SYNとARP RequestとNetBIOS-SSN SYN，サービス妨害攻撃の監視項目としてTCP SYN，意図しない活動の監視項目としてIRC RequestとSMTP Requestを管理する．PINC項目はプロセス情報取得時に動作中の全Process Image Name集合を意味し，文字列としてDBにおけるprimary keyとして管理する．

$$PINC = process1, process2, process3, \cdots$$

マルウェアは実行ファイル形式として存在するため，感染した場合はPINCに変化を及ぼす．Outbound Port項目はOutbound通信先Port番号を管理する．特定PINCにおいて出現しないPort宛のパケット発信が検出可能となる．Microsoft-DS SYN及びNetBIOS-SSN SYN項目はPort139，445宛のSYNパケット発信数を管理する．これらパケットはWindows上でのファイルやプリンタ共有サービスであるSMBサービス稼働ホストの探索に使用される．SMBサービスの脆弱性を利用するマルウェアによる活動が検出可能となる．ARP Request項目はARP Requestパケット発信数を管理し，ネットワーク接続端末をARP Probeにより探索する活動を検出する．TCP SYN項目はPort139,445宛以外のTCP SYNパケット発信数を管理し，TCPによるDoS攻撃活動を検出する．IRC Request項目はIRC Requestパケット発信数を管理し，IRCサーバへの接続要求であるIRC Request中のNICK，USERHOST，MODE，JOIN，PONG，PASSコマンド送信を検出する．SMTP Request項目は，SMTPプロトコル中に送出されるSMTP HELOコマンドの発信数を管理する．SMTP HELOコマンド発信数を監視することでメール送信数を取得し，過剰なSMTP通信によるSPAM送信を検出する．各発信数の項目は出現した各PINCにおける発信パケットの時間平均発信数(times/sec)とする．監視エージェント及びセンタにおいて同様形式のDBを有する．

4.2 状態数収束性と重複性

DBはPINCをprimary keyとして構築し，状態判定に使用する．DB肥大化及び解析時間を考

表 1: 収集ログ概要

User	OS	Hours	User Events	Process Conditions	IP Outbound Conns
A	WinXP	199	3244	144	3284434
B	WinXP	173	6501	141	3658593
C	WinXP	316	3550	168	1383329
D	WinXP	176	3378	129	2102319
E	WinXP	460	8104	195	5520876
F	WinXP	188	7025	144	1128136
G	WinXP	148	2071	192	485195
H	WinXP	134	4264	136	1324940
I	WinXP	287	7201	181	3024824

慮するとPINC出現数は時間経過に伴い収束する必要がある．一般にユーザの使用状態は類似し，かつ時間経過と共に出現する新たな状態数は収束すると推測される．しかし，新規出現するプロセスは定常化する傾向が強く，随時追加されるためにPINC出現数に収束性は現れない．そこで取得したPINCからプロセス間に関連関係を持つプロセス集合を除去する．マルウェアが起動するプロセスはユーザ毎の他プロセスとの関係性を考慮しないため，PINCの一部除去で感染状態の特徴は損なわれない．関連関係を有するプロセス集合を「同時起動状態にあるプロセス集合」と定義する．関連関係を各プロセス毎のプロファイルで管理し，PINC中に関連プロセス集合を全て含有する場合，関連状態としてPINCから除去する．実環境9台で収集したログ概要を表1，本手法適用によるUserAにおける関連状態除去前後のPINC出現数を図2，User間のPINC重複性を表2に示す．関連状態除去した場合，数十時間経過後に出現状態数が収束性を持つ．表2中のDeltaはPINC間のプロセス差分数を意味し，Delta1はプロセス1個分の差分値を持つ状態まで含めた重複性を意味する．Delta2まで含めた場合，状態重複率は約87%となる．

4.3 Update Software Database

一般的なアプリケーションは自動更新プログラムによりソフトウェア更新やセキュリティ脆弱性の修正を行う．自動更新が実行される場合，新たなPINCが出現する．監視対象が増加した場合，更新プログラムによる出現状態を収集し，異常検知に使用できるが現在実環境ログを多く取得していないため，専用機による状態収集を行う．ユーザ使用頻度の高い更新プログラムを専用収集機で定期実行し，出現状態を収集する．対象となるプログラムはMicrosoft Update，Adobe Update，Google Update，Apple Software Updateとする．各プログラムを自動更新に設定し，収集した状態は随時センタDBに追加する．

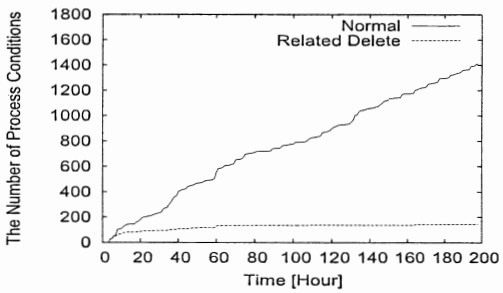

図 2: 実環境における状態出現数

5 異常検知手法

異常検知部は，監視エージェントとセンタ内に存在する．異常検知部は，検知対象情報を自身のDB内の各PINCにおけるエンティティ項目値を使用して，正常・異常判定を行う．エージェントは自身で発生したPINCのみ管理する

表 2: 実環境における状態重複性

User	Delta0[%]	Delta1[%]	Delta2[%]
A	63.09	71.81	80.54
B	65.75	78.77	92.47
C	55.81	70.35	88.37
D	50.00	64.93	86.57
E	50.51	59.69	91.33
F	50.69	65.97	88.19
G	44.27	57.29	85.94
H	51.43	63.57	82.86
I	47.31	60.75	90.86
Avg.	53.2	65.9	87.4

エージェント DB を用いて，以下の条件で状態判定を行う．

1. 完全一致する PINC が存在する．
2. 項目 1 を満たす状態においてエンティティ項目が全て正常である．

1，2 両条件満たした場合に正常とし，自身の DB を更新する．Outbound 通信先 Port 番号は全ての Port 番号が DB に格納された Port 情報に含有される場合，他項目は DB 格納値以下の場合に正常状態とする．異常の場合，センタへ検知対象情報を送信して解析を依頼する．センタはエージェントから通知を受信後，検知対象情報をセンタ DB を用いて，以下の条件で状態判定を行う．

1. (a) 完全一致する PINC が存在する．
   (b) 項目 (a) を満たす状態においてエンティティ 1 項目以下が異常である．
2. (a) 指定した状態間距離内の PINC が存在する．
   (b) 項目 (a) を満たす状態においてエンティティ項目が全て正常である．

$$Threshold = \alpha \times AverageNum. \quad (1)$$

状態間距離は検知対象 PINC と DB 内 PINC 間のプロセス差分数を指す．1，2 いずれかの条件を満たした場合，正常とする．Outbound 通信先 Port 番号はエージェント異常検知と同様，他項目は式 (1) で定義される閾値以下となる場合に正常状態とする．センタは状態判定後，エージェントへ判定結果を通知する．正常の場合，エージェントは自身の DB を更新する．異常の場合，検知対象情報とセンタ DB で管理する状態間で最小異常項目数の状態における異常エンティティ項目をエージェントに通知する．

6 実装

攻撃対象となる OS の多くが Windows であることから本システムの WindowsXP 用プロトタイプを構築した．プロトタイプはアプリケーション層で動作するが，システム情報を取得するにはカーネル内に組み込んで動作させることが望ましい．

状態検出部は，監視対象となるホスト上で動作し，psloglist[3] を用いて定期的に PINC 監視を行う．psloglist で動作する全てのプロセス名を文字列として取得し，UNICODE 順にプロセス名を並び替えて MD5 値を取得する．定期的に MD5 値を取得し，異なる MD5 値が出現した場合，PINC 変化検出として状態解析部へ通知する．また状態検出部は同時に WinPcap[4] を必要とする WinDump[5] を用いて，特定の PINC 出現中における全ての通信パケットを保存する．状態解析部は，状態検出部から通知を受け，PINC 及びパケットの解析を行う．PINC 解析には各種プロファイルを用い，PINC から固定，重複，関連状態を除去する．固定状態を「一日間常駐のプロセス集合」と定義し，管理プロファイルを用いて PINC から除く．関連状態除去は PINC における各プロセスの関連状態プロファイル中に含まれる集合が PINC 内に全て含有される場合，関連状態として除去する．除去後に各プロセス毎のプロファイルを更新する．PINC 解析後，tshark[6] によるパケット解析を行う．固定プロセス状態から発信されるパケットの送信先 Port 番号を考慮し，「一日間全時間帯で発信する Outbound Port 番号」を固定 Port 状態として Outbound Port 項目から除去する．全エンティティ項目の解析後，状態検出部に通知して保存したパケットを削除し新たなパケット保存を開始する．その後，検知対象情報として PINC とエンティティ項目値を異常検知部へ通知する．異常検知部はエージェント・センタ共に，検知対象情報を自身の持つ DB を用いて状態判定を行う．

7 システム評価

異なるユーザが実環境使用する 9 台の Windows PC に本システムを導入し実環境評価を行う．表 1 中のログ取得時間は実働時間である．また，2 台の PC で構築するテスト環境でマルウェア 21 検体の侵入検知評価を行う．システムの主要目的はマルウェア感染時に早期検知し，誤検知数を最小化する点にある．実環境評価では False Positive Rate(以下 FPs)，侵入検知評価では感染時における False Negative Rate(以下 FNs) 及び検知に要する時間を評価する．

7.1 実環境評価

9 台の Windows PC で 20 日間以上，本システムを導入し実環境評価を行う．収集ログ概要は表 1 に示す．表 1 の PC8 台と Update Software Database で構築したセンタ DB を用いて評価する．PINC 取得時間間隔 1 秒，$\alpha = 1$，解析対象 PINC 間距離 2 とした場合の評価結果を表 3，時間経過に伴う誤検知数の評価結果を図 3，4 に示す．解析対象 PINC 間距離は表 2 より状態重複率が 90%近いため，Delta2 までを解析対象とした．DB 構築及び実環境評価時にマルウェアによる攻撃脅威が存在しないと仮定し，異常検知を全て誤検知とみなす．FPs は実環境 PC9 台で平均 1.45%となり，誤検知数は時間経過と共に漸近的減少する．60 時間経過後の FPs は平均 0.42%となり，更に監視対象を監視する時間が経過するに従って，誤検知数は飛躍的に減少する．異常状態として検出した状態には仮定通り，マルウェア感染状態の存在を確認できなかった．ログ取得環境のセキュリティが堅固であることが要因であると考えられる．誤検知数は数十時間経過後には，多くて数日に一回検知される程度となり，実環境使用可能な水準となる．

7.2 侵入検知評価

テスト環境を構築し，マルウェア侵入時の検知評価を行う．テスト環境は検体を動作させる検体実行機とサーバ機の 2 台で構成し，外部ネットワークには一切接続していない．実行機には Vmware[7] 等の仮想 OS 環境を使用する方法もあるが，検体種によっては解析回避するために正常動作しないため，実環境で感染時ログを取得した．検体実行機の OS は Windows XP，サーバ機の OS は Linux を用いる．感染時の外部通信応答のため，DNS，IRCd 及び Postfix をサーバ機で稼働する．DNS は全ての名前解決依頼に対して自身の IP アドレスを返答する．本システムを検体実行機に導入し，長期間稼動後に検体を動作させてログを取得する．評価対象としてマルウェア 21 検体を用意し，McAfee による解析を行った [8]．検知結果を検体科名ごとに分類すると W32/Sdbot.worm が 5 検体，W32/Gbot.worm が 3 検体，W32/Korgo.worm が 2 検体，W32/Bobax.worm が 2 検体，W32/Gael が 1 検体，Unknown が 8 検体となる．Unknown とは McAfee で検知不可能な検体である．表 1 中の全 PC9 台と Update Software Database で構築したセンタ DB を用いて，侵入検知評価を行う．PINC 取得時間間隔 1 秒，$\alpha = 1$，解析

表 3: 実環境誤検知評価

User	FPs[%]	FP After 60 hours	FPs After 60 hours[%]
A	2.343	5	0.274
B	0.815	9	0.448
C	2.213	12	0.511
D	1.184	2	0.096
E	1.037	60	0.830
F	0.797	10	0.226
G	2.897	16	1.228
H	1.220	0	0
I	0.569	9	0.181
Avg.	1.45		0.42

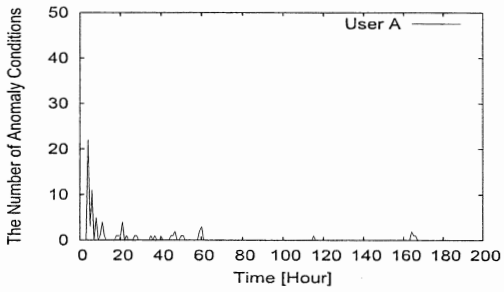

図 3: 実環境における誤検知数 1

表 4: マルウェア検知要因 1

科名 (Family Name)	検体数	Process	Outbound Port	SMTP	IRC
W32/Sdbot.worm	5	1	4		2
W32/Gbot.worm	3		3		
W32/Korgo.worm	2	1	1		
W32/Bobax.worm	2	1	1		
W32/Gael	1				
Unknown	8	4	4		2

表 5: マルウェア検知要因 2

科名 (Family Name)	ARP	Microsoft-DS	NetBIOS-SSN	TCP
W32/Sdbot.worm	2			
W32/Gbot.worm		3	3	
W32/Korgo.worm		1		
W32/Bobax.worm				1
W32/Gael		1		
Unknown				

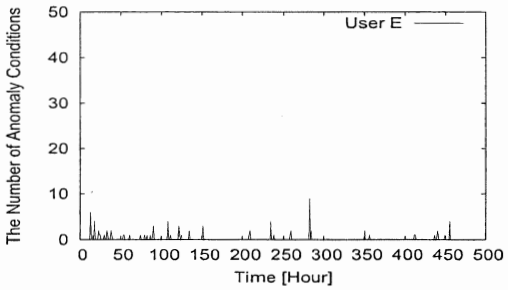

図 4: 実環境における誤検知数 2

表 6: マルウェア実行時検知結果

科名 (Family Name)	検体数	検知数	平均検知時間 [秒]
W32/Sdbot.worm	5	5	1.20
W32/Gbot.worm	3	3	1.00
W32/Korgo.worm	2	2	319.50
W32/Bobax.worm	2	2	1.50
W32/Gael	1	1	2.00
Unknown	8	8	1.25
All	21	21	

対象 PINC 間距離 2 として本システムを約 100 時間稼動後，マルウェアを動作させて検知評価を行った結果を表 6，各検体科名毎の検知要因を表 4，5 に示す．この結果，21 検体を全て検知したため FNs は 0%となった．McAfee で検知不可能な Unknown8 検体も実行直後に全て検知した．W32/Korgo.worm1 検体で感染直後には検知できず検知時間が大きくなっている．これは感染直後に PINC 変化を検知不可能であったことが原因であるが，後に出現した PINC において異常状態として検知を確認した．検知までにかかる時間が大きいのは，検体実行機はログ収集中無操作状態であるため，新たな PINC が出現しないことが原因である．実環境では PINC 変化は頻繁に発生するため，検知に時間を要しないと考えられる．これより PINC の変化を検知不可能な検体も侵入時に異常状態として検知できた．そのため，本システムはルートキットによってシステム情報を隠ぺいする検体による異常状態の検知が可能であると考えられる．

8 むすび

本稿では，ホームネットワークの家電機器で出現するプロセスの集合を複数の監視対象項目を用いて状態判定し，異常検知するシステムを提案した．特にマルウェア侵入を検知対象とする監視項目を設定した結果，誤検知数は状態収集時間の経過と共に漸近的に減少させることができる．また，ルートキットなどシステム情報を隠ぺいする検体を含めた未知の検体を検知することが可能となる．

参考文献

[1] 増田 耕一, 牧野 義樹, 中田 潤也, 丹 康雄, “家電製品使用における異常状態のモデル化とその検知手法の提案,” UBI2006，No.54, pp.43-50，2006.

[2] 相川 恵, “ホームネットワークの障害診断に関する研究,” 北陸先端科学技術大学院大学情報システム学専攻修士論文, 2007.

[3] PsTools, http://technet.microsoft.com/en-us/sysinternals/bb896649.aspx.

[4] WinPcap, http://www.winpcap.org/.

[5] WinDump, http://www.winpcap.org/windump/.

[6] tshark, http://www.wireshark.org/docs/man-pages/tshark.html.

[7] VMware. VMware Player, http://www.vmware.com/products/player/.

[8] McAfee. Virus Information, http://us.mcafee.com/virusInfo/default.asp.