Vol.2013-DPS-154 No.53
Vol.2013-CSEC-60 No.53
2013/3/15

SaaF：Sandbox as a File の提案

可児潤也[†1]　米山裕太[†1]　加藤岳久[†2]　間形文彦[†3]　勅使河原可海[†4]
佐々木良一[†5]　西垣正勝[†2]

近年，攻撃対象を特定の組織や組織内の個人に限定した標的型攻撃による脅威が深刻化している．その多くは，標的に対してゼロデイの脆弱性をつく添付ファイル付きのメールを送り，受信者が添付ファイルを開いてしまうことで発症する．そこで，本論文では，VM によって添付ファイルごとに個別のサンドボックスをあてがい，利用のたびにそのサンドボックスを使い捨てる方式を提案する．本方式は，カプセル化コンテンツを安全に実行させる環境としても有用である．

SaaF : Sandbox as a File

JUNYA KANI[†1] YUTA YONEYAMA[†1] TAKEHISA KATO[†2]
FUMIHIKO MAGATA[†3] YOSHIMI TESHIGAWARA[†4] RYOICHI SASAKI[†5]
MASAKATSU NISHIGAKI[†2]

Threat of APT (Advanced Persistent Threat) that is targeting a particular organization or individual is becoming more serious. Many of them send their target an e-mail with an infected attachment that exploits a zero-day vulnerability, and infected by the recipient would open the attachment. This paper proposes a countermeasure using VM in which a separate-and-disposal sandbox is allocated to each attachment. The proposed scheme, Sandbox as a File (SaaF), is also useful as a safe environment to execute the encapsulated contents.

1. はじめに

近年，攻撃対象を特定の組織や組織内の個人に限定した標的型攻撃（Advanced Persistent Threat，APT）[1]による脅威が深刻化しており，実際に組織の知財情報や個人情報を搾取されるという事件が起きている[2]．これらの攻撃は，組織または個人が一定のセキュリティ対策を行っていても被害を受けていることが確認されている．

その多くは，標的に対してゼロデイの脆弱性をつく添付ファイル付きの標的型攻撃メールを送り，受信者がこの添付ファイルを開いてしまうことで発症する．標的型攻撃メールは非常に巧妙で，送信アドレスが標的の知人のものに偽装されているだけでなく，メール本文の内容についても正規のメールと見分けがつかない．そのため，受信者の多くは，添付ファイルを開いてしまう場合が多い．最近は金銭目的で悪事を働く不正者が増加していると言われており[3]，組織の機密ファイルを窃取するために標的型攻撃が利用されるケースも少なくない[4]．そのため，標的型攻撃に対する早急な対策が求められている．

†1 静岡大学大学院情報学研究科
Graduate School of Informatics, Shizuoka University
†2 静岡大学創造科学技術大学院
Graduate School of Science and Technology, Shizuoka University
†3 NTT セキュアプラットフォーム研究所
NTT Secure Platform Laboratories
†4 創価大学大学院工学研究科
Graduate School of Engineering ,Soka University
†5 東京電機大学未来科学部
School of Science and Technology for Future Life, Tokyo Denki University

標的型攻撃に対する対策としては，予防接種アプローチ[5]，なりすましメールの検知手法[6,7]，アンチウイルスソフトによる添付ファイルの検査などが提案されている．しかし，ソーシャルエンジニアリングを巧みに利用した標的型攻撃の場合には，どんなに予防接種アプローチを取ったとしても，騙されてしまうユーザは少数ではないだろう．また，標的型攻撃の場合は巧妙な偽装メールが使用されることが多いため，なりすましメールの検知についても限界がある．更に，標的型攻撃においてはゼロデイの脆弱性が突かれることが多いため，添付ファイルをアンチウイルスソフトで検知する方法もその効果は限定的である．このほか，入口対策，出口対策に関する研究や製品が報告されているが，著者らの調べた限りにおいては，それらのいずれの方法も検知漏れ，誤検知，利便性低下が課題とされており，現在のところ，十分に効果的な標的型攻撃対策は存在していないというのが実情である．

そこで本論文では，VM を利用し，ファイルごとに個別のサンドボックスをあてがい，利用のたびにそのサンドボックスを使い捨てる方式を提案する．提案方式においては，ホスト OS 上にはファイルが静的な状態（クローズされた状態）でのみ存在することになるため，ホスト OS が感染ファイルに汚染される可能性を排除することが可能となる．提案方式は VM を多用する方式となっているが，近年の PC の性能は十分に高いため，パフォーマンスの低下は限定的であると考えられる．以上より，提案方式は，どのようなファイル形式のファイルが用いられたとしても，かつ，どのような脆弱性が突かれたとしても有効であり，更に利便

性も損なうことのない標的型攻撃対策となっている．

更に，提案方式は，ファイルを安全に実行するための汎用的な環境としても利用することが可能である．典型的には，カプセル化コンテンツを安全に実行させる環境として有用である．

以下，2 章に標的型攻撃についてとその対策を紹介し，3 章で提案方式を説明する．4 章で関連研究との差異を示したのち，5 章でカプセル化コンテンツの実行環境としての適用例を詳述する．最後に 6 章でまとめを示す．

2. 標的型攻撃とその対策

2.1 標的型攻撃の攻撃段階

標的型攻撃（APT）には複数の攻撃段階がある[1]．以下に各段階を説明する．

① 攻撃準備段階

標的の組織に関係ある組織へ攻撃を行うなどして，標的の組織に関する情報を得る．

② 初期潜入段階

準備段階で得た情報を利用し，組織内の特定のメールアドレスに，関係者を装ってメールを送信する．添付ファイルにはゼロデイの脆弱性をつくファイルを添付されている．標的が添付ファイルを開いてしまうことで感染してしまう．

③ 攻撃基盤構築段階

感染した端末に RAT（Remote Administration Tool）をインストールさせ，攻撃者の C&C サーバからのバックドアを確立する．

④ システム調査段階

RAT が内部のシステムの情報を収集する．ここで得た情報を基に，目的とする情報へアクセスする．この段階は，数週間から数か月にわたる場合もある．

⑤ 攻撃最終目標の遂行段階

RAT が組織の重要な情報を窃取し，攻撃者（C&C サーバ）へ送信する．送信する情報を独自に暗号化して送ったり，ログを削除するなど，仮に侵入が発見されてしまった場合でも，その後の解析を困難にするような操作が行われる場合もある．また，取得した組織の内部情報（組織内のアカウント情報等）を利用して更なる攻撃を加える場合もある．

標的型攻撃は，メールでなく USB メモリ等の外部メディアを利用した攻撃も考えられる．Stuxnet の事例[8]はその典型例であり，「隔離ネットであれば安全である」という神話は今日では成り立たない．

2.2 標的型攻撃への対策と課題

2.1 節で述べた各段階における標的型攻撃対策手法とその課題点を述べる．

① 攻撃準備段階

本段階ではソーシャルエンジニアリングに類する攻撃手法が主流になると考えられる．そのため，標的となるユーザが本段階で，技術的なアプローチによって対策することはほぼ不可能であると考えられる．

② 初期潜入段階

本段階では標的型攻撃メールの検知に関するアプローチが考えられる．標的型攻撃メールの検知については，予防接種アプローチ[6]や，電子メールの特徴情報を利用した標的型攻撃メール検知手法[7,8]が提案されている．

前者は，標的型攻撃メールそのものを病原体と仮定し，模擬標的型攻撃メールを「予防接種」として利用することで，「人」に免疫反応を引き起こし，ユーザの注意力を高める試みである．後者は，メールヘッダや本文の中に含まれる送信者固有の特徴情報を利用することで，標的型攻撃と思われるメールに対して警告を出す．しかし，標的型攻撃メールは非常に巧妙であり，正規のメールと見分けがつかないため，これらの方式では不十分であると場合も多い．後者には，アンチウイルスソフトを利用してメールの添付ファイルを検査する方法も含まれるが，標的型攻撃の場合はゼロデイの脆弱性が用いられることも多いため，アンチウイルスソフトは有効ではない．

標的型攻撃メールにおいては，JavaScript を含む PDF やマクロ機能を含む Microsoft office 文書に関する脆弱性が利用されることが多いため，JavaScript やマクロ機能をディスエイブルにするという対策は，ある程度有効に機能する．また，「防人」と呼ばれる製品は，外部からのメールに添付されてきたファイル一旦画像ファイルにして受信者に送信することによって，JavaScript やマクロ機能のディスエイブルと同等の効果をもたらす．しかし，これらの対策は，ファイルの機能が制限される分，ユーザの利便性が損なわれる．また，「防人」の場合は，受信者が希望すればオリジナルファイルをダウンロードすることができる．そのため，画像化した場合に本物と区別のつかなくなるような不正な文書ファイルが使用された場合には，感染の危険性が否定できない．

③ 攻撃基盤構築段階

本段階においては，RAT のインストール検知，または RAT と C&C サーバとの通信検知が考えられる．

しかし，標的型攻撃に用いられる RAT の多くは未知のマルウェアが利用されると考えられるため，一般的なアンチウイルスソフトによって RAT を検知することは困難である．また，RAT の多くは，HTTP や HTTPS などの正規の通信を装って C&C サーバとの通信を行うため，通常の方法ではこの通信を捉えることは難しい．

文献[9]では，組織と外部ネットワークの間にプロキシを設置し，組織内からはじめてアクセスする URL に対し

て，CAPTCHA やワンタイムパスワードを利用して RAT の C&C サーバへのアクセス要求を遮断しようという方法が提案されている．しかし，正規プロセスの中にもバックグラウンドで外部のサーバへの通信を行うものが存在するため，誤検知の可能性も残る．また，文字型の CAPTCHA を解読するマルウェアに関する報告[10]もなされており，RAT が CAPTCHA を突破しないとも限らない．

④　システム調査段階

システムを調査する段階では，RAT は長い時間をかけて潜伏調査する場合が多く，攻撃の検知につながるような顕著な特徴は得られにくいと考えられる．

⑤　攻撃最終目標の遂行段階

本段階では，いわゆる出口対策としての機密情報漏洩対策の利用が考えられる．

例えば，組織内の機密情報を暗号化しておくことによって，暗号化データが窃取されたとしても，機密情報の漏洩を防ぐことができる．ただし，標的型攻撃の場合は，暗号化データを復号するための鍵情報まで盗まれる状況となる場合も多い．

DLP (Data Loss Prevention) [11]のように，外部に送信されるデータに機密情報が含まれるかどうかをチェックし，含まれる場合はその通信を遮断するという製品も存在する．しかし，「機密情報」を完全に定義しきれないことも多いため，DLP のようなルールベースの情報流制御が十分に機能しない場合も考えられる．

文献[13]の方法は，発信ファイルに対して送信者による検印を要請することで，検印のないファイルの送信をプロキシサーバによってフィルタアウトする．しかし，今後この方式が普及すれば，RAT が検印してから外部へ送信するような方式をとることが考えられる．

3. Sandbox as a File

3.1 コンセプト

2 章で示したように，著者らの調べた限りにおいては，現在のところ，十分に効果的な標的型攻撃対策は存在していないというのが実情である．

そこで本論文では，VM を利用し，ファイルごとに個別のサンドボックスをあてがい，利用のたびにそのサンドボックスを使い捨てる方式を提案する．提案方式を，SaaF（Sandbox as a File）と名付ける．

SaaF においては，ホスト OS 上の任意のファイルがオープンされる度に，新たに専用サンドボックス（使い捨て VM）が生成され，当該ファイルデータがそのサンドボックス（使い捨て VM 上のゲスト OS）にコピーされた上で，そのサンドボックス内でのみ当該ファイルが開かれる．そして，ファイルが閉じられた瞬間に，このサンドボックス（使い捨て VM）も消滅し，当該ファイルデータがホスト OS にコピーバックされる．これによって，ホスト OS 上にはファイルが静的な状態（クローズされた状態）でのみ存在することになるため，ホスト OS が感染ファイルに汚染される可能性を排除することが可能となる．

サンドボックス内で感染ファイルが開かれた場合には，そのサンドボックス（使い捨て VM 上のゲスト OS）は汚染されることになるが，ファイルのクローズとともに当該サンドボックスも消滅するため，汚染 VM が定常的にホスト OS 上に存在するようなことはなく，汚染 VM を通じて 2 次感染が発生する懸念も抑えられる．

SaaF は VM を多用する方式となっているが，近年の PC の性能は十分に高いため，パフォーマンスの低下は限定的であり，ユーザの利便性も保たれるものと考えられる．

以上より，SaaF は，どのようなファイル形式のファイルが用いられたとしても，かつ，どのような脆弱性が突かれたとしても有効であり，更に利便性も損なうことのない標的型攻撃対策となっている．

更に，提案方式は，（標的型攻撃対策になるだけでなく，）ファイルを安全に実行するための汎用的な環境としても利用することが可能である．典型的には，カプセル化コンテンツを安全に実行させる環境として有用である．（これについては，5 章で詳述する．）

3.2 基本スキーム

以下では，PDF ファイルが添付されているメールを例に採り，SaaF を具体的に説明する．

現在のファイルシステムで PDF ファイルを開く場合は，①AdobeReader のプロセスが作られ，②添付 PDF ファイルが AdobeReader プロセスに渡されることによって，③PDF ファイルが AdobeReader によって表示される．そして，ユーザが PDF ファイルをクローズすると，④AdobeReader プロセスが kill される．

これに対し，SaaF の環境下で PDF ファイルを開く場合は，①VM が 1 台立ち上がり，②VM 上に AdobeReader のプロセスが作られた上で，③添付 PDF ファイルが VM のゲスト OS にコピーされる．更に，④VM にコピーされた PDF ファイルが VM 上の AdobeReader プロセスに渡され，⑤VM 上の AdobeReader によって PDF ファイルが表示される．そして，ユーザが PDF ファイルをクローズすると，⑥VM 上の AdobeReader プロセスが kill されるとともに，⑦VM 上の PDF ファイルがホスト OS のファイルシステムにコピーバックされる．更に，⑧VM 自体も kill される．

ここで，SaaF における VM は，AdobeReader プロセスの専用シェルの形で実装され，ユーザからは，ホスト OS 上に AdobeReader プロセスが表示されているように見える．当然，VM に対するユーザ操作は透過的になっており，ユーザは「VM 上の AdobeReader プロセス」を「ホスト OS

Vol.2013-DPS-154 No.53
Vol.2013-CSEC-60 No.53
2013/3/15

上の AdobeReader プロセス」のように操作することができる．すなわち，SaaF におけるユーザビリティの低下はほとんどない．現在のファイルシステムの概念図と SaaF の概念図を，図 1，図 2 に示す．

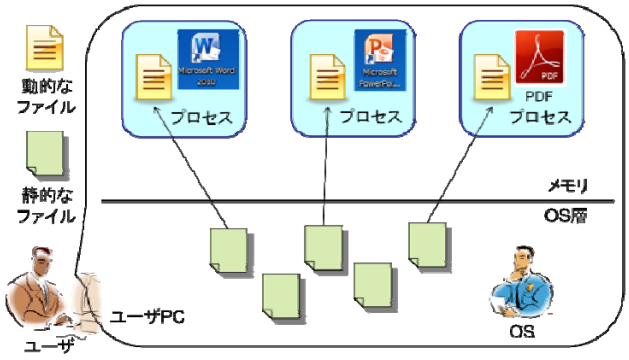

図 1：現行のファイルシステム

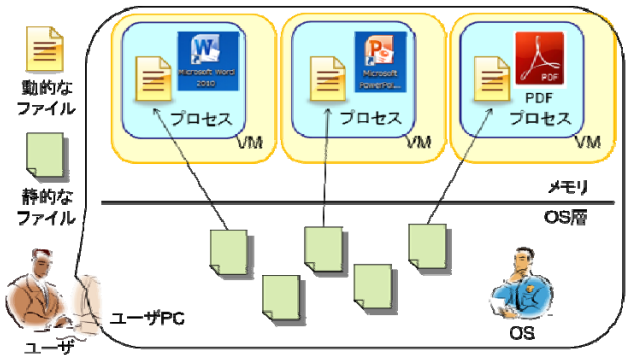

図 2：SaaF による VM 型ファイルシステム

もし，PDF ファイルに Adobe の脆弱性を攻撃する悪意ある動的コンテンツが仕掛けられていた場合，既存のファイルシステムにおいては（ホスト OS 上の）AdobeReader プロセスが感染し，ホスト OS および（ホスト OS 上の）ファイルシステムに被害が及ぶことになる．一方，SaaF においては，VM（ゲスト OS）上の AdobeReader プロセスが感染し，ゲスト OS および（ゲスト OS 上の）ファイルシステムは攻撃を受けるが，ホスト OS および（ホスト OS 上の）ファイルシステムに被害が及ぶことはない．なお，SaaF の場合は，PDF ファイルをクローズした時点で，VM 上の当該 PDF ファイルがホスト OS のファイルシステムにコピーバックされるが，この時点においては当該 PDF ファイルは静的な状態（オープンされていない状態）であるため，ホスト OS および（ホスト OS 上の）ファイルシステムは安全である．

3.3 実装形態

現時点では，SaaF の実装形態として，以下の 2 つを考えている．

(1) As-needed Sandbox

特定のファイル（例えばメールの添付ファイルや，あやしいダウンロードファイル）のみをサンドボックス上で実行する場合の実装形態可能である．例えば，SaaF システム用アイコンをデスクトップに常駐させておき，怪しいと思われるファイルをオープンする場合のみ，ユーザが当該ファイルを SaaF アイコンにドラッグ＆ドロップする．この操作によって使い捨て VM が立ち上がり，当該ファイルが VM 上でオープンされる．当該ファイルをクローズすると，VM ごと kill される．

(2) Any-time Sandbox

すべてのファイルに対して，SaaF の適用を強制する場合の実行形態である．ホスト OS 上で，任意のファイルをオープンする場合には，必ず使い捨て VM が立ち上がり，当該ファイルが VM 上でオープンされる．（例えば，ファイルのアイコンをダブルクリックすると，自動的に使い捨て VM が立ち上がり，当該ファイルが VM 上でオープンされる．）当該ファイルをクローズすると，VM ごと kill される．

4. 関連研究と考察

4.1 単一 VM との違い

ホスト OS を守るという目的だけを考えた場合，VM を 1 つだけ用意して，怪しいファイル（例えば添付ファイル）を VM 上で実行させるような方法を採ることも可能である．しかし，1 つの VM を使い続けることになると，不正ファイルによって VM がマルウェアに感染してしまった後に，その VM 上で別のファイルを開こうとした際に，別のファイルも二次感染してしまう．これを防ぐためには，SaaF のようなファイル単位での VM の使い捨てが必要となる．

4.2 「防人」との違い

「防人[12]」は，添付ファイルつきのメールを受信すると，添付ファイルを画像形式に変更して受信者へ送信する．よって，元の添付ファイルがマルウェアであったとしても，受信者には，画像ファイルのみが送信されるため安全である．しかし，「防人」においては，受信者が動的なコンテンツ（例えば WORD や EXCEL のマクロ機能）の利用を望む場合には，サーバから元のファイルをダウンロードしなければならない．すなわち，ユーザの利便性が損なわれる．また，受信者が画像ファイルを確認し，正規の添付ファイルであると判断した場合には，受信者はサーバから元のファイルをダウンロードして，これを使用するであろう．このため，画像化した場合に本物と区別がつかなくなるように，不正者が巧妙に不正ファイルを作成した場合には，「防人」では対応できない．一方，SaaF であれば，ユーザの利便性と安全性の両者を維持することが可能である．

4.3 GoogleDocs との違い

Google によって提供されている GoogleDocs [14]等のクラウドファイルシステムを利用することによっても，SaaF と類似の効果が得られると考えられる．すなわち，メールに添付されたファイルをクラウドサービス上で開くことによって，クライアント側のユーザ PC を守るという対策が可能である．しかし，標的型攻撃では，社外秘扱いの重要ファイルに見せかけた形で添付ファイルが送られてくる可能性も大いにあり得る．このような場合は，GoogleDocs 等のクラウドサービス上で添付ファイルを開くことが躊躇われる．また，一般的なクラウドサービスはブラウザを介しての利用となるため，マルウェアによってはクライアント PC 側のブラウザが感染する可能性がある．SaaF は，ユーザ個人のエンド PC 上での対策となっているため，機密情報ファイルの扱いも問題とならない．また，SaaF の場合は，VM 上のブラウザが感染しても，ホスト OS には被害が及ぶことはない．

5. SaaF 上でのカプセル化コンテンツの実行

5.1 情報カプセル化とその課題

コンテンツ保護の分野で，情報をカプセル化して流通させるカプセル化技術に関する研究が報告されている[15][16][17][18]．情報カプセルとは，流通対象のデジタルコンテンツとその操作のためのプログラムを一体化したもので，このプログラムにコンテンツの利用制御機能を持たせることによってコンテンツの著作権保護を実現することなどが，その典型的な利用例である．カプセル化コンテンツの概念図を図 3 に示す．

しかし，利用側からみると，カプセル化コンテンツの利用は，外部から受信したプログラムをユーザ自身のマシンで実行することを意味し，大きなセキュリティ懸念につながる．すなわち，カプセル化コンテンツは，マルウェア（特にトロイの木馬）の温床となり得るという深刻な問題が存在している．

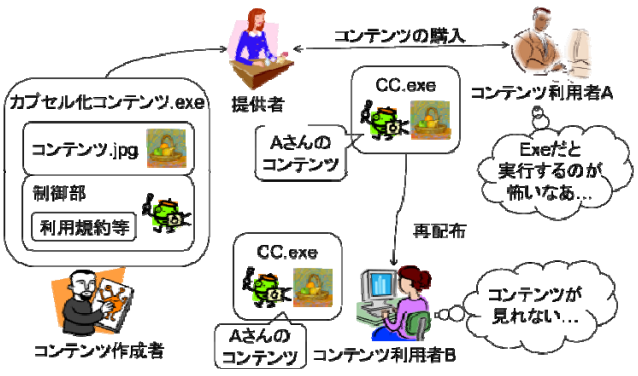

図 3：カプセル化コンテンツ

5.2 SaaF の適用

SaaF 上でカプセル化コンテンツを利用する場合を考える．SaaF においては，カプセル化コンテンツが VM 上で実行されることになる．よって，もしカプセル化コンテンツがマルウェアだった場合でも，マルウェアが感染するのは VM であり，ホスト OS に影響はない．これにより，ユーザはカプセル化コンテンツを安全に扱うことが可能となる．このように，SaaF は，カプセル化コンテンツに限らず，様々な実行形式ファイルの取り扱いにおける安全性の確保を可能とする．

6. まとめ

標的型攻撃（APT）メールに対する対策として，個々のファイルに対して各々の使い捨てサンドボックスをあてがう SaaF（Sandbox as a File）の仕組みを提案した．今後は SaaF の実装と評価を進めていく．

参考文献

1) IPA,「新しいタイプの攻撃」の対策に向けた設計・運用ガイド，改定第 2 版
2) 原発事故情報が外部流出…ＰＣのウイルス感染で http://www.yomiuri.co.jp/net/security/ryusyutsu/20120611-OYT8T00832.htm ,(2012.06)
3) ［ウイルス・不正アクセス］イタズラ型から金銭目的型へ http://response.jp/article/2012/01/06/167969.html , (2012.01)
4) 「国内企業の約 1 割が標的型攻撃の被害」とシマンテックが説明，http://itpro.nikkeibp.co.jp/article/NEWS/20111130/375257/, (2011.11)
5) 山口健太郎，小宮山功一郎，内田勝也：ユーザへの予防接種というアプローチによる標的型攻撃対策，情報処理学会研究報告，CSS2008，pp.385-390 (2008.10).
6) 吉岡孝司，片山佳則，津田宏，森永正信，深澤亮太：電子メールの特徴情報を用いた標的型メールへのクライアント対策技術の提案，情報処理学会研究報告，2012-CSEC-58，pp245-252 (2012.07).
7) 梅田昂翔，上原雄貴，水谷正慶，武田圭史：電子メールヘッダの特徴情報を用いた標的型攻撃の検知，情報処理学会研究報告，CSS2010，pp109-114 (2010.10).
8) Stuxnet の事例から学ぶ「新しいタイプの攻撃」 http://ascii.jp/elem/000/000/621/621423/ (2011.08)
9) 北澤繁樹，河内清人，桜井鐘治：標的型攻撃におけるマルウェア通信の検知と対策，情報処理学会研究報告，SCIS2012, (2012.01).
10) J.Yan,A.S.E.Ahmad: Breaking Visual CAPTCHAs with Naïve Pattern Recognition Algorithms, 2007 Computer Security Applications Conference, 2007.
11) ITpro, DLP とは, http://itpro.nikkeibp.co.jp/article/Keyword/20081028/317943/ ,2008
12) ネットエージェント，防人, http://www.netagent.co.jp/sakimori.html
13) 榊原裕之，桜井鐘治： APT におけるファイル漏洩の防止について， SCIS2013, (2013.01)
14) Google, GoogleDocs, https://drive.google.com/
15) 明石修，森保健治，寺内敦：カプセル化機構を用いた情報流通方式：FleaMarket，情報処理学会論文誌, Vol.39, No.2, pp.458-465 (1998.02)
16) 加賀美千春，森賀邦広，塩野入理，櫻井紀彦：コンテンツ流通における自立管理を目的としたカプセル化コンテンツ Matryoshka,

Vol.2013-DPS-154 No.53
Vol.2013-CSEC-60 No.53
2013/3/15

情報処理学会研究報告, マルチメディア通信と分散処理,
pp.99-104 (2000.03)
17) 谷口展郎, 難波功次, 塩野入理：情報カプセル流通における利用者システム保護, 情報処理学会研究報告, マルチメディア通信と分散処理, pp.103-108 (2001.02)
18) 山田孔太, 木下広揚, 森住哲也：情報フィルタと情報カプセルによる著作権・所有権保護システム, 情報処理学会研究会報告, 2006-CSEC-81, pp.43-50 (2006.07)