RAT サーバの動作を遠隔操作者と同じ操作画面で観測する方法 その 2

髙橋 佑典† 小林 大朗† 陳 悦庭†
小山 大良† 金井 文宏† 吉岡 克成† 松本 勉†

†横浜国立大学
240-8501 神奈川県横浜市保土ケ谷区常盤台 79-7
{takahashi-yusuke-pw, kobayashi-masaaki-ny}@ynu.jp, f9190yuki@gmail.com,
{oyama-taira-vn, kanei-fumihiro-tv}@ynu.jp, {yoshioka, tsutomu}@ynu.ac.jp

あらまし 近年問題となっている標的型攻撃ではRAT (Remote Administration Tool, Remote Access Trojan) が利用されることが多い. これに対し, 我々はRATを遠隔操作する様子を攻撃者と同じ操作画面でリアルタイム観測する手法を提案した. しかしながら, 評価実験で検証を行ったRATが2種類と少なく, またRATがサポートする様々な遠隔操作機能について観測可否の詳細な検証ができていなかった. そこで本稿では, 6種類のRATに対して実験を行なうことで提案手法の有効性を示し, また遠隔操作の内容ごとに観測可否を検証する. 加えて提案手法の拡張として, 攻撃者との通信内容を蓄積・再生することで効率的に事後観測を行なう方法を示す.

Observing RAT server's behavior using its client GUI Part2

Yusuke Takahashi† Masaaki Kobayashi† YuehTing Chen†
Taira Oyama† Fumihiro Kanei† Katsunari Yoshioka† Tsutomu Matsumoto†

†Yokohama National University
79-7 Tokiwadai, Hodogaya, Yokohama, Kanagawa 240-8501, JAPAN
{takahashi-yusuke-pw, kobayashi-masaaki-ny}@ynu.jp, f9190yuki@gmail.com,
{oyama-taira-vn, kanei-fumihiro-tv}@ynu.jp, {yoshioka, tsutomu}@ynu.ac.jp

Abstract Remote Administration Tool or Remote Access Trojan (RAT) is often used in targeted attacks. In order to monitor the RAT's behavior controlled by a remote attacker, we have proposed a method to observe RAT server through its client GUI. However, we had tested our method with only few RATs and we did not verify what type of operations could be observed by our method. In this paper, we show the validity of our method with six kinds of RAT and examine which remote operations are observable. In addition, we show that we can also accumulate and reproduce the communication between RAT and the attacker and analyze the attacker's behavior afterward for forensic purpose.

1 はじめに

近年, 特定の企業や組織を狙った標的型攻撃が脅威となっている. この標的型攻撃には, しばしば RAT (Remote Administration Tool, Remote Access Trojan) が使用される. RAT は攻撃者に対象ホストのリアルタイム遠隔操作機能を提供するツールであり, 操作対象のホスト上で動作する RAT サーバと, RAT サーバを遠隔操作するための RAT クライアントからなる. 近年の RAT クライアントは, 直感的な遠隔操作を行なえるような GUI が提供されていることが多く, その使いやすさから様々な攻撃に悪用されている.

標的型攻撃への対策が重要であることは広く認識

されているが，有効な対策が採られず多くのインシデントが発生している．この原因として，攻撃側の振る舞いや動向に関する情報不足が挙げられる．我々は攻撃者の RAT を利用した不正活動の実態を把握することが重要であると考え，攻撃者の行動監視手法として RAT クライアントの GUI を用いる手法を提案した[6, 8]．PoisonIvy と Cerberus という 2 種類の RAT を用いた検証実験において，「マシン情報の取得」と「Remote Shell」等の操作に関して，攻撃者が操作する RAT クライアントと同様の画面を監視用 RAT クライアントの画面に表示させられることを示した．また，RAT にはビルド時にパスワードを設定し，RAT サーバ・クライアント間のセッション確立時にパスワード認証を行なうものがあるため，Virustotal[1]に投稿されている RAT サーバ検体からパスワード抽出実験を行い多くの検体からパスワードを抽出できることを示した．しかしながら，実験対象とした RAT が 2 種類と少なく，RAT がサポートする様々な遠隔操作機能のうち，いずれが提案手法により観測可能であるか詳細な検証ができていなかった．

本稿では，提案手法によるリアルタイム監視の評価実験を 6 種類の RAT に対して行なうと共に様々な遠隔操作機能について観測の可否を検証する．また，RAT サーバと攻撃側との通信内容を蓄積・再生することでリアルタイム監視だけでなく効率的な事後観測も行えることを示す．事後観測は，リアルタイム監視を行えない場合や，動的解析の解析結果を後から調査したい場合などに有用であり，RAT サーバと攻撃者との通信の再現をパケット単位でステップ実行することで，任意のスピードで操作内容を分析できる上，有意な操作が行われていないアイドル状態をスキップし効率的に分析を行えるという利点がある．さらに，本手法を回避する方法についても攻撃者の立場から考察する．

2 関連研究

標的型攻撃のように実態が十分に知られていない脅威に対して，攻撃者の振る舞いや動向を把握することでその後の対策に活かそうという研究開発が近年活発に行われている．

論文[2]では標的型攻撃に関連する様々な要素に基づいたグルーピング結果から攻撃者の実態を明らかにすることを目的とし，メールを起点に行われる標的型攻撃（以下，標的型メール）を行なう攻撃者をグルーピングする手法を提案している．まず標的型攻撃が，攻撃対象の組織を不正プログラムに感染させる「攻撃準備」，重要情報の検索や他の端末へ侵入を試みる「攻撃開始」，重要情報の外部漏洩等を行なう「目的遂行」の 3 つから成るとし，それぞれの段階に適した要素をグルーピングの項目としている．「攻撃準備」では標的型メールのメールヘッダ，「攻撃開始」で標的型メールには添付された不正プログラムや二次検体のファイル情報やレジストリ操作，「目的遂行」では接続先の IP アドレスやドメイン名などをグルーピング項目とし，実データに適用している．

論文[3]では，標的型攻撃のシナリオに沿った動的解析を行える解析環境を提案している．複数の Windows マシンやプロキシサーバなどの企業を模した被害環境や，C&C サーバや ExploitKit を設置した WWW サーバなどの攻撃環境を用意し，解析者が標的型攻撃のシナリオ再現を行いやすくなっている．また，被害環境の構成は柔軟に変更できるため，あらゆる組織を想定した実験・分析を行なうことができる．

総務省は，標的型攻撃の攻撃手法の解析が困難である点や，攻撃を受けたあとの対処が確立されていないことなどを受け，官公庁や企業等向けの演習を実施している[4]．この演習は「サイバー攻撃解析・防御モデル実践演習の実証実験」[7]の一環であり，サイバー攻撃の解析では標的型攻撃等のサイバー攻撃情報の迅速かつ効率的な収集と正確な解析を，防御モデルの検討ではサイバー攻撃被害の把握などを目的としている．また，製品[5]は，標的型攻撃に対応したセキュリティアプライアンスである．標的型攻撃を防御・検知するアプローチとして，既に攻撃者が組織内のネットワークで活動しているという仮定を立てている点が特徴としてあげられる．その仮定に基づいて提供している機能が，実リソースを保護する仮想ネットワーク，攻撃者をおびき寄せるおとりエージェント，攻撃者の意図を掴むためのアクティビティ記録である．これらでは，標的型攻撃を行なう攻撃者の意図を把握することが目的のひとつとなっており，標的型攻撃を行なう攻撃者の行動特性を把握することが重要であることは明らかである．

3　提案手法

本章では，活動中の RAT サーバの挙動を RAT クライアントの操作画面を用いて観測するリアルタイム監視と，RAT サーバの通信のキャプチャデータを用いた事後観測について説明する．リアルタイム監視と事後観測の概要図をそれぞれ図 1，2 に示す．リアルタイム監視は，監視対象の RAT サーバが動作する監視対象ホスト，監視用の RAT クライアントが動作する監視用ホスト群，監視対象ホストと攻撃者の中間にあり，監視対象ホストから攻撃者への通信を観測し，監視用ホスト群に対して同様の通信を送信するためのプロキシホストからなる．一方，事後観測は，監視対象の RAT サーバが動作する監視対象ホスト，RAT サーバから攻撃者の RAT クライアントへ送られる通信を観測・キャプチャするプロキシホスト，キャプチャデータを読み込み，再現を行う通信再現用ホスト，監視用の RAT クライアントが動作する監視用ホスト群からなる．

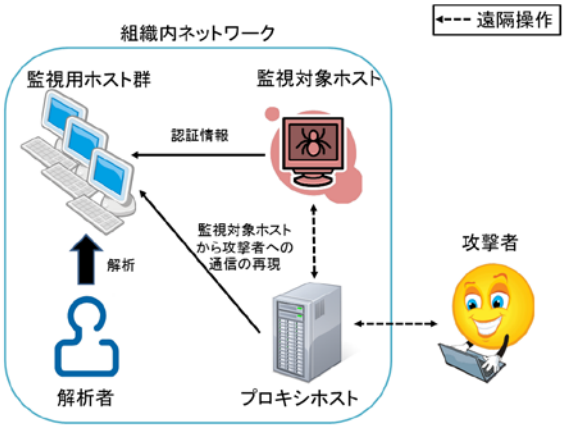

図 1　リアルタイム監視の概要

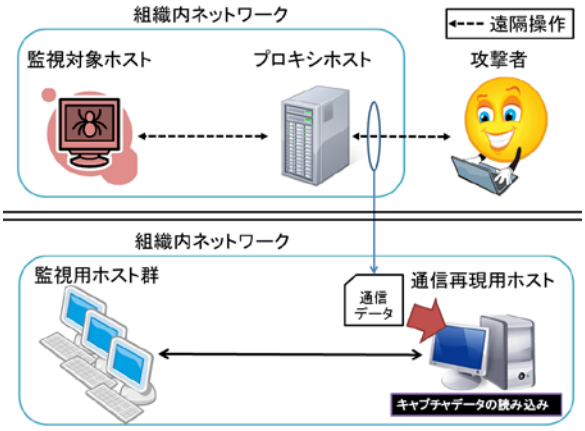

図 2　事後観測の概要

3.1　認証情報の抽出

まず，監視対象の RAT サーバから，攻撃者の RAT クライアントとの接続に必要な情報の抽出を行なう．ここで抽出する情報は，アクセス先の IP アドレス/ドメイン名，ポート番号，パスワードの 3 つである．但し，認証を行なわないタイプの RAT についてはパスワードが存在しないため，それ以外の情報を抽出する．これらの認証情報の抽出には様々な方法が考えられる．認証情報が RAT サーバの実行可能ファイルの特定箇所に埋め込まれている場合は，シグネチャベースのパターンマッチにより抽出が可能である．RAT サーバがパッカーなどで暗号化されている場合は，メモリフォレンジックツール等を利用して実行時にメモリに展開されるデータから抽出を行なう．上記以外にもより高度な解析により認証情報を取得することができるが，論文[6, 8]の評価実験では，パターンマッチとメモリフォレンジックツールによる抽出のみを行っている．

3.2　監視用 RAT クライアントの設定

監視用ホスト上で監視用 RAT クライアントを実行する．次に，3.1 節で抽出したパスワードやポート番号を監視用 RAT クライアントに適用し，接続待機状態にする．なお，PoisonIvy クライアントのように 1 つのウィンドウ内で操作項目を選択して使用するタイプの GUI をもつ RAT クライアントでは，選択していない操作項目については監視用ホストの GUI に反映されない．そこで，監視したい操作項目分だけ監視用ホストを用意し，各監視用ホスト上で監視対象の操作項目を選択しておく．全ての監視用ホストに同様の通信を送信することで複数の項目の同時監視を行なう．

3.3.1　リアルタイム監視

プロキシホストで監視対象ホストから攻撃者の RAT クライアントへの通信を観測し，同様の通信を監視用ホスト群に送信する．RAT サーバを実行するとプロキシホストが監視用 RAT クライアントに同様の通信を送信し，監視用 RAT クライアントとプロキシホストのセッションが確立する．

3.3.2 事後観測

事後観測では，まず RAT サーバの通信のキャプチャデータを用意する．通信再現用ホストでこのキャプチャデータを読み込み，RAT サーバから攻撃者の RAT クライアントへ送られる通信をパケット単位で監視用ホスト群に送信する．

3.4 再接続対応

何らかの理由で監視用ホスト群とのセッションが切れた場合は，再接続を試みる．セッション確立後にパスワードによる認証を行なう RAT については，認証時の通信を保存しておき，この通信を再送することで再接続を行なう．再接続に成功した場合，その後の RAT サーバの通信を監視用ホスト群に送信する．

4 観測可能な操作

本章では提案手法による観測が可能な操作について定性的な考察を行う．一般に RAT クライアントと RAT サーバは図 3 のような処理により，リアルタイム遠隔操作を実現していると考えられる．まず，①RAT クライアント上で攻撃者が操作を行い，②操作に対応した内部処理が行われ，③操作を反映した通信(リクエストと呼ぶこととする) が RAT クライアントから RAT サーバに送られる．④RAT サーバは受信したリクエストに応じた処理を実行し，⑤その結果をレスポンスとして RAT クライアントに送信する．最後に⑥RAT クライアントは RAT サーバからのレスポンスを受信し，その内容を GUI に表示する．但し，全ての操作に対してリクエストとレスポンスの通信が発生するわけではなく，表 1 のように様々なパターンが存在すると考えられる．

提案手法では RAT サーバから攻撃者の RAT クライアントへ送信されるレスポンスを観測し，監視用 RAT クライアントに同様の通信を送ることで監視を行なう．そのため，提案手法で監視が行える操作は RAT クライアント上で攻撃者が行なった操作に対してリクエストが発生し，さらにこれを受信し対応する処理を行った RAT サーバがレスポンスを返信するものに限られる．また，レスポンス通信が発生したとしても，RAT クライアントの操作画面に反映されないものは結果を GUI 上で把握することができない．これより，提案手法で観測可能な操作は，表 1 のパターン 5 の，「操作に対するレスポンス通信が発生し，かつ操作結果が画面に反映される操作」であるといえる．パターン 4 は，レスポンスはあるものの画面に操作結果が反映されないため監視が行えない．

また，表 1 のパターン 3 は RAT クライアントの操作に対して RAT サーバが操作結果を送信しないものである．このような操作はレスポンスが発生しないため提案手法による監視ができない．例えば，ファイルマネージャ機能によって RAT サーバが動作しているマシン上に新しくフォルダを作成する場合などがこのパターンにあたる．また RAT によっては，フォルダ作成時に成功可否を表示しない．しかし，ファイルリストの取得などの別の操作で先のフォルダ作成の成功可否を把握することは可能である．

そのため，操作結果を送信しないパターンでも，別の操作によって操作結果を把握することができるので，パターン 3 にあたる操作は観測ができる余地が残っている．

パターン 1･2 は RAT クライアントに表示されている操作結果をローカルに保存するなど，ローカルの RAT クライアントで完結してしまう操作である．このパターンはそもそも通信が発生しないため，提案手法

表 1 RAT の操作処理のパターン

No	パターン	備考
1	①→②	ローカルの操作(画面反映なし)
2	①→②→⑥	ローカルの操作(画面反映あり)
3	①→②→③→④	リモートの操作(レスポンスなし)
4	①→②→③→④→⑤	リモートの操作(レスポンスあり･画面反映なし)
5	①→②→③→④→⑤→⑥	リモートの操作(レスポンスあり･画面反映あり)

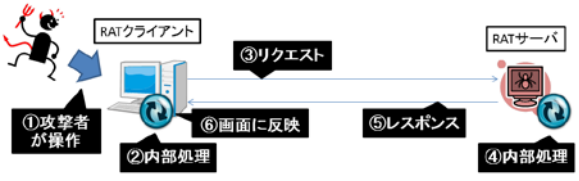

図 3 RAT の処理の流れ

による観測が原理的に不可能である.

5 実験

本章では, RAT サーバビルダで作成したテスト検体を用いた実験について記述する. まず, リアルタイム監視を 6 種類の RAT に対して行い, 操作ごとの監視可否を調べる(実験 1). 次に, 実験 1 と同様の操作を事後観測し, 成功可否を調べる(実験 2). 最後に, 実験 1･2 で監視･観測した操作について調査し考察する. 実験では, 我々が RAT クライアントと RAT サーバビルダを入手することができ, 実験のために動作させることができた 6 種類の RAT (表 2)を評価対象とした. Gh0stRAT と DarkCometRAT はオプションでパスワードを設定できるため, 5.4 節でパスワード抽出の可否も調査する.

表 2 評価対象の RAT

名称	プラットフォーム	パスワード	その他
PoisonIvy	Windows	有	
Cerberus	Windows	有	
BandookRAT	Windows	無	一部平文通信
Gh0stRAT	Windows	有※	
DarkCometRAT	Windows	有※	Windows XP SP1,3 では動作せず.
AndroRAT	Android	無	
有※…オプションで設定可能			

5.1 リアルタイム監視

RAT サーバビルダを用いて作成したテスト検体を監視対象ホスト上で実行し, 提案手法によるリアルタイム監視を行なう. RAT に実装されている各操作を攻撃者ホストの RAT クライアントから使用し, 監視の可否を調査する.

1 台の実マシン上に, 監視対象ホスト, 監視用ホスト, 攻撃者ホストをそれぞれ仮想マシンとして用意し, 攻撃者ホストは仮想ネットワーク NW1 に, 監視対象ホストと監視用ホストは同一の仮想ネットワーク NW2 に属するようにする. このとき, NW1 と NW2 はプロキシホストを介して接続する. 実験環境のネットワーク構成を図 4 に示す.

プロキシホストで各ホストの通信の監視を行い, RAT サーバから攻撃者の RAT クライアントへ送られたパケットを観測した場合は, 即座に同様のパケットを監視用ホストへ送信する. プロキシホストはホストマシン上で動作する iptables と自作の Python スクリプトにより実装した.

まず, 表 2 に示した各 RAT について, それぞれの RAT サーバビルダを用いてテスト用検体を用意した. これらを監視対象ホスト上で実行して攻撃者ホスト上の RAT クライアントで操作を行い, 監視用ホスト上の RAT クライアントで各操作が監視できるか実験を行なった. また, 4 章で述べた表 1 のパターン 5 に該当する操作を実験対象とした.

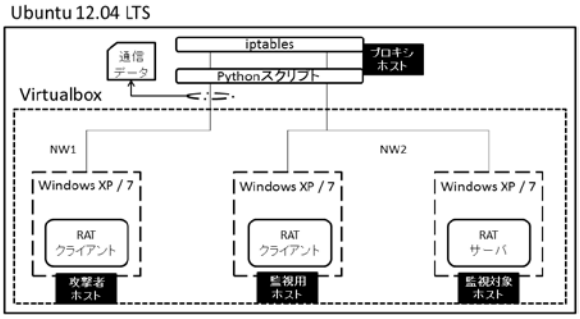

図 4 リアルタイム監視環境

4 章の考察の通り, 提案手法による観測ができる可能性のある操作は表 1 のパターン 5 であるため, RAT に実装されている主な機能のうちパターン 5 にあたるものを表 3 にまとめる. BandookRAT は画面の反映がされない場合があり, 複数回同じ操作を行った結果を載せている. 表 3 の通り, ほとんどの操作について観測が可能であることがわかった. 同様の操作でも RAT によって監視できる場合とできない場合があり, 監視可能な操作に差が見られた. 特に, RAT サーバが動作しているマシンをリモートデスクトップのように操作することができるスクリーンキャプチャ機能については, 攻撃者の RAT クライアントに表示されている画面と完全に同じ画面を反映させることができたのは Gh0stRAT のみであった.

5.2 事後観測

5.1 節で作成したテスト検体の操作通信をキャプチャし, そのキャプチャデータを用いて同様に事後観測の可否を調査する.

事後観測実験は 5.1 節の実験環境を用いて行なう. まず, プロキシホスト上で RAT サーバから RAT クライアントへ送信される通信をキャプチャし, 操作ごと

に保存する. RATにはTCPセッション確立後に認証を行なうものがあるため, 検証したい操作を行なう度に再接続を行い, 操作通信をキャプチャした. 次に, 監視用ホスト上で RAT クライアントを接続待機状態にし, 通信再現用ホストでキャプチャデータを読み込み, 監視用ホスト上の RAT クライアントへ送信する. 実験環境のネットワーク構成を図 5 に示す. 通信再現用ホストは, 自作の Python スクリプトにより実装した.

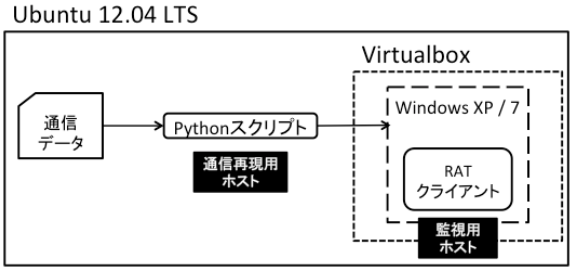

図 5 事後観測環境

表 3 と同様に機能ごとの観測結果を表 4 に示す. リアルタイム監視の結果とほぼ同様の結果となっているが, BandookRAT ではリアルタイム監視で監視できなかったが, 事後観測では観測できている操作があった. 反対に DarkCometRAT では, リアルタイム監視で監視可能だったが, 事後観測では観測不可となっている操作があった.

5.3 観測可能性について

5.1, 5.2 節の実験で監視/観測に失敗した操作について, その原因を調査する. 失敗のケースを大きく分けると以下の 2 つのケースに分類できる.

A.監視用 RAT クライアントとプロキシホストのセッションが切断される
B.攻撃者の RAT クライアントの画面が監視用 RAT クライアントに反映されていない

ケース A は, PoisonIvy のスクリーンキャプチャ機能, Cerberus のスクリーンキャプチャ機能とキーロガー, ファイルマネージャ機能のダウンロード/アップロード操作が該当する. その他の失敗した操作はケース B にあたる.

また, 同様の操作でも, RAT によって処理パターンが異なるものがあることがわかった. 一例として, PoisonIvy と Cerberus のレジストリマネージャ機能が挙げられる. Cerberus のレジストリマネージャ機能によってリモートホストに新たにレジストリを登録した場合, 操作結果を示すメッセージが表示されるが, PoisonIvy で同様の操作を行っても画面には操作結果を示す表示がされない. そのため, PoisonIvy のレジストリ登録操作は攻撃者でさえ, 操作直後に結果を把握することはできない. しかし, レジストリ一覧を取得する操作を行なうことでレジストリ登録の成功可否を知ることが可能であり, 登録に成功していた場合は提案手法でも, この一覧取得の操作による画面の差分でレジストリ登録を把握することができる.

5.4 パスワード抽出について

セッション確立時に認証を行なう RAT に対しては, 予め監視対象の RAT サーバに設定されているパスワードを把握しておく必要がある. Gh0stRAT と DarkCometRAT はオプションでパスワードを設定可能であり, 論文[6, 8]では扱っていなかったため, メモリフォレンジックツールである Volatility[9]を用いたパスワード抽出を試みた. その結果, パスワードを設定した Gh0stRAT と DarkCometRAT の RAT サーバ実行時にメモリ展開したコードの中からパスワード等の設定情報が含まれる箇所を特定することができ, これを抽出することに成功した.

6 考察

-リアルタイム監視/事後観測 原理的に本手法で監視/観測が可能な操作は表 1 のパターン 5 の操作であるが, パターン 5 の操作でも監視/観測のできない場合があることがわかった. 同様の操作であるにもかかわらず, スクリーンキャプチャ機能のように RAT によって成功可否が大きく違う操作があり, この原因は RAT の実装方法にあると考えられる. その他の操作についても RAT の種類によって成功可否に差があり, 本手法は RAT の実装方法に強く影響を受けるといえる.

また, 提案手法による監視/観測が原理的にできない操作のうちパターン 3 については, 本手法による監視/観測可能性が残っている. 5.3 節で述べた PoisonIvy のレジストリ登録操作がこのパターン 3 にあたる. 本手法は監視用 RAT サーバからの通信をプロキシホストで破棄しているため, RAT サーバのレ

表 3 操作ごとのリアルタイム監視結果

操作項目	PoisonIvy	Cerberus	BandookRAT	Gh0stRAT	DarkCometRAT	AndroRAT※
マシン情報取得	◎	◎	◎	◎	△	×
ファイルマネージャ	△	○	△	○	△	
レジストリマネージャ	◎	○	○		×	
プロセスマネージャ	◎	◎	○		△	
ウィンドウマネージャ	◎	◎	◎		×	
サービスマネージャ	◎	◎	○		×	
キーロガー	◎	×	×		×	
スクリーンキャプチャ	×	×	○	◎	△	×
リモートシェル	◎	◎	◎		×	

◎…すべての機能で監視可能
○…ほぼすべての機能で監視可能
△…一部の機能で監視可能
×…監視不可
※…Windows 向けの RAT に共通する項目でまとめたため，共通機能が少ない

表 4 操作ごとの事後観測結果

操作項目	PoisonIvy	Cerberus	BandookRAT	Gh0stRAT	DarkCometRAT	AndroRAT※
マシン情報取得	◎	◎	◎	◎	△	×
ファイルマネージャ	△	○	△	○	×	
レジストリマネージャ	◎	○	○		×	
プロセスマネージャ	◎	◎	○		△	
ウィンドウマネージャ	◎	◎	◎		×	
サービスマネージャ	◎	◎	◎		×	
キーロガー	◎	×	◎		×	
スクリーンキャプチャ	×	×	○	◎	△	×
リモートシェル	◎	◎	◎		×	

◎…すべての機能で監視可能
○…ほぼすべての機能で監視可能
△…一部の機能で監視可能
×…監視不可
※…Windows 向けの RAT に共通する操作項目でまとめたため，共通機能が少ない

スポンスを受動的に受信しており，パターン 3 の操作結果を把握するためには攻撃者のリクエストを待つしかないが，監視用 RAT クライアントからリクエストを送りレスポンスを能動的に取得できるようにすることで，パターン 3 の操作結果も反映できる可能性がある．

-本手法の回避方法 提案手法による監視/観測は，攻撃者ホストからの操作に対する RAT サーバのレスポンスを監視用 RAT クライアントにも送ることで実現している．この方法で監視/観測が成功する RAT は，ステートレスに実装されていると考えられる．そのため，RAT クライアントをステートフルに実装されてしまうと提案手法による監視/観測がうまくいかない．AndroRAT は操作のリクエストに ID を付与し，自身が発行した ID 以外からのレスポンス通信を受け付けない実装になっていたため，すべての操作について監視/観測に失敗したと考えられる．

DarkCometRAT についてもほとんどの操作で監視/観測に失敗しており，これもステートフルな実装がなされていたと思われる．

-観測時に RAT クライアントが入手できない場合 提案手法では監視対象の RAT サーバに対応する RAT クライアントを入手しているという前提がある．しかし，監視対象の RAT サーバに対応する RAT クライアントが外部公開していないなどの理由で入手で

きない場合は十分に考えられる. このような場合は, 以下の対応を取ることによって, いずれ対応する RAT クライアントが入手できた際に提案手法を適用できるよう備えることができる.

・認証情報を取得しておく
・動的解析によって通信のキャプチャデータを取得しておく

標的型攻撃は攻撃者自身が遠隔操作するというリアルタイム性の高い攻撃であり, 対応するRATクライアントが入手できた後に改めて解析を開始しても攻撃者側からの操作が発生しなかったり, 攻撃者の RAT クライアントと繋がらないケースが考えられるため, RAT クライアントが手にはいらない場合は別の手法を採用し, 通信データをキャプチャして提案手法の事後観測に備えることが得策といえる.

7 まとめと今後の課題

RAT の各操作について提案手法の成功可否を調査し, 提案手法で観測可能な操作を明らかにした. また, 提案手法による監視を回避されてしまう場合と, 観測時に RAT クライアントを入手できない場合について考察した.

実験にはテスト用にビルドした検体を用いたため, 実際に攻撃者が操作する RAT クライアントと接続を行い, その操作を観測することが今後の課題として挙げられる. また, RAT クライアントの操作に対してレスポンス通信が発生しないケース (表 1 のパターン 3) において, 監視用 RAT クライアントから能動的に状態確認を行う方法の詳細な検討と自動化についても今後の課題である.

謝辞 本研究の一部は, JSPS 科研費 24680006 の助成により行われた.

参考文献

[1] VirusTotal, http://www.virustotal.com/
[2] 北条 孝佳, 松浦 幹太, "標的型攻撃における攻撃者のグルーピング手法, " SCIS2014, 4C1-2, 2014.
[3] 津田 侑, 神薗 雅紀, 遠峰 隆史, 安田 真悟, 三浦 良介, 宮地 利幸, 衛藤 将史, 井上 大介, 中尾 康二, "標的型攻撃のシナリオ再現環境の構築, " 情報処理学会研究報告.CSEC, 2014-CSEC-65 巻, 18 号, pp.1-6, 2014-05-15.
[4] 総務省, "総務省｜「実践的サイバー防御演習 (CYDER)」の実施, " http://www.soumu.go.jp/menu_news/s-news/01ryutsu03_02000057.html (最終閲覧日:2014/08/16)
[5] Shadow Networks, " Shadow Networks™ Advanced Threat Deception, " http://www.shadownetworks.com/wp-content/uploads/Shadow-Networks_Bro_Final.pdf (最終閲覧日:2014/08/16)
[6] 髙橋 佑典, 小林 大朗, 陳 悦庭, 米持 一樹, 吉岡 克成, 松本 勉, "RAT サーバの動作を遠隔操作者と同じ操作画面で観測する方法, "コンピュータセキュリティシンポジウム 2013 論文集, 2013 巻, 4 号, pp.279-286, 2013-10-14.
[7] 総務省, "政府における情報セキュリティ政策の取組について, " http://www.soumu.go.jp/main_content/000274855.pdf (最終閲覧日:2014/08/16)
[8] 情報・物理セキュリティ拠点, ”攻撃者によるRATの遠隔操作をリアルタイム監視, ” http://ipsr.ynu.ac.jp/ratmon/
[9] volatility - An advanced memory forensics framework - Google Project Hosting, https://code.google.com/p/volatility