1W-09

レジリエントサーバの改良と性能評価

佐野　史和†　　岡本　剛†　　Idris Winarno‡　　畑　良知‡　　石田　好輝‡

神奈川工科大学†　　豊橋技術科学大学‡

1. はじめに

IoT 時代の幕開けとともに，可用性の高いインターネットサービスの需要が高まっている．しかしながら，従来の高可用性を実現する運用技術であるフォールトトレランス設計では，サイバー攻撃を想定していない場合が多い．この問題を解決するために，本研究では，生物学的多様性に触発された，サイバー攻撃に頑健なアタックレジリエントサーバを設計・実装を行ってきた [1) 2)]．しかし，文献 [2)] では，耐故障機能によりサービスのダウンタイムが伸びる結果となった．この原因は，仮想マシンの制御のたびに認証が必要な制御コマンドを使用していたためと考えられた．

そこで，本稿では，アタックレジリエントサーバにおける仮想マシンの制御をコマンドで制御する Bash スクリプトから，仮想マシンを制御可能な API で制御する Perl スクリプトに変更し，各仮想マシンに対する認証セッションを保持することで，切り替えにかかる時間を短縮した．本実験では，変更前と変更後の性能の違い及びサービスのダウンタイムを比較評価する．

2. サイバー攻撃

代表的なサイバー攻撃には，エクスプロイト，DoS 攻撃，情報漏えいがある．本研究ではエクスプロイト及び DoS 攻撃を対象とし，これらのサイバー攻撃から保護することとする．ただし，DoS 攻撃の 1 つである，UDP ベースの増幅攻撃は，サーバ側で対処できないため，本研究は，この攻撃を除外する．この攻撃の解決方法には，サービスを提供するサイトを増やすほかに，インターネットサービスプロバイダやコンテンツデリバリネットワークなどが提供する DoS 攻撃対策のサービスや，Anycast を使用する方法がある．

3. プロトタイプシステム

サービスの継続性を評価するため，DNS サービスを提供するプロトタイプシステムを構築した（図 1）．プロトタイプシステムは，管理アプライアンス，ハイパーバイザ，NAS（Network Attached Storage）から構成される．管理アプライアンスは，VMware vCenter Server Appliance 6.0 である．ハイパーバイザには，VMware vSphere 6 Enterprise Plus に含まれる VMware ESXi 6.0 を使用する．NAS には OpenMediaVault を用いた．

3.1. 物理サーバの構成

図 2 に示す通り，物理サーバは複数の仮想マシンから構成され，いずれか 1 つの仮想マシンがサービスを提供する．それぞれの仮想マシンは，実装の異なるサーバアプリケーションと異なる OS によって構成される．脆弱性は，仕様上の脆弱性と共有ライブラリの脆弱性を除けば，サーバアプリケーションの実装に依存するため，構成の異なるすべてのサーバアプリケーションが同様の脆弱性の影響を受けることはこれまでほとんどない．

プロトタイプシステムでは，ハイパーバイザ上に 2 台の Windows と 1 台の Linux の仮想マシン

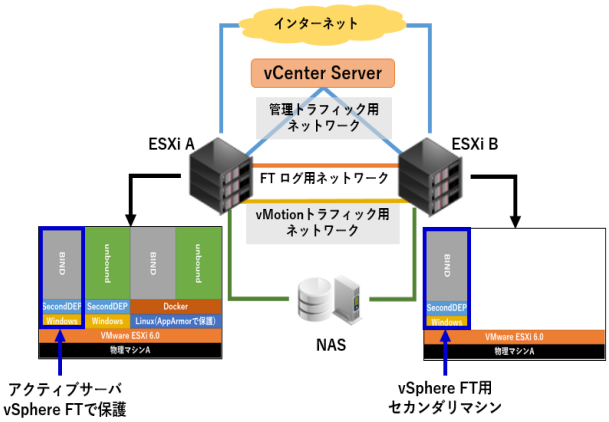

図 1 アタックレジリエントサーバの構成

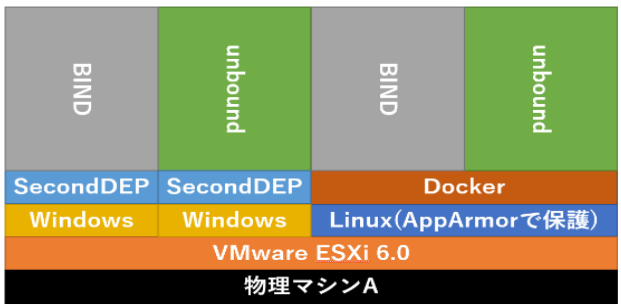

図 2　物理サーバの構成

Improvement and its evaluation of a resilient server
†Fumikazu Sano, Takeshi Okamoto
Kanagawa Institute of Technology
‡Idris Winarno, Yoshikazu Hata, Yoshiteru Ishida
Toyohashi University of Technology

を使用する．Windows の仮想マシンには ISC BIND 9 または unbound が稼働する．Windows のサーバアプリケーションに対するエクスプロイトは SecondDEP により検知し[3]，DoS 攻撃は名前解決のタイムアウトにより検知する（3.2 節参照）．Linux の仮想マシンには，サーバアプリケーションをアプリケーションレベルで仮想化するため，Docker を使用して ISC BIND 9 と unbound を稼働させる．Docker の利点は，ハイパーバイザの仮想マシンと比べてコンテナを迅速に切り替えられ，メモリやストレージなどのリソースの消費が少ない点である．Linux のサーバアプリケーションに対するエクスプロイトは AppArmor により検知し，DoS 攻撃は Windows と同様である．

3.2. 仮想マシンの切り替え制御

仮想マシンの切り替え制御は，管理アプライアンス内のサービスモニタと仮想マシンチェンジャーにより行われる．サービスモニタは，サービスの稼動状態を定期的にチェックする．プロトタイプシステムは，名前解決のタイムアウトを 500 ミリ秒に設定し，連続で 3 回タイムアウトしたとき，サービスが停止したと判断する．このとき，サービスモニタは仮想マシンチェンジャーに仮想マシンの切り替えを指示する．仮想マシンチェンジャーは，待機している仮想マシンに切り替える．同時に，次の切り替えに備えて，サスペンド状態の仮想マシンをバックグラウンドでレジュームし待機させる．

これらの機能は vSphere SDK for Perl により実装した．vSphere SDK は仮想マシンを制御する API を提供する．vSphere SDK により，各仮想マシン及び ESXi とのセッションの保持が可能となり，セッション確立のオーバーヘッドを減らせる．

3.3. フォールトトレラントシステム

フォールトトレラントシステムは，図 1 に示した通り，2 つ以上の物理サーバから構成される．このフォールトトレラントシステムは VMware vSphere FT の機能を利用する．この機能により，サービスを提供している物理サーバまたはネットワークに障害が発生した場合，仮想マシンを停止することなく，セカンダリサーバに移行しサービスを継続できる．

4. パフォーマンステスト

仮想マシンチェンジャーの実装について，Bash と Perl の違いを評価するため，サイバー攻撃による仮想マシン切り替え時における，それぞれのダウンタイムを計測した．

図 3 にこれまでの結果（Bash スクリプト）と

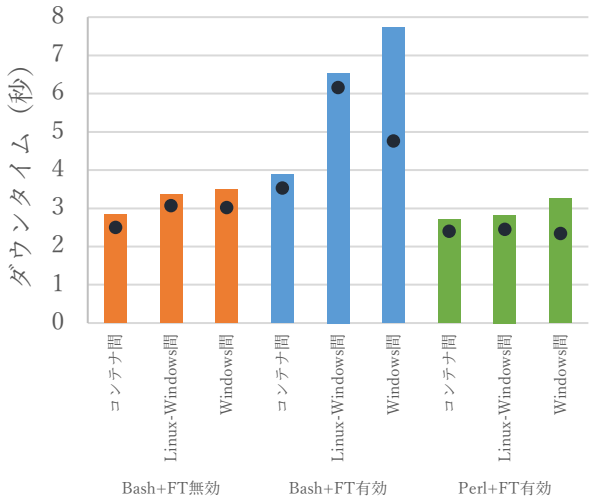

図 3 ダウンタイムの比較

本研究の結果（Perl スクリプト）を示す．棒グラフはダウンタイムの最大値，プロットはその平均値である．計測結果は 10 回の試行から得られたダウンタイムの平均値と最大値である．

計測結果から，すべての切り替え時においてフォールトトレランスを有効にしても，ダウンタイムを平均で 2.5 秒程度下げることができた．また，各切り替えのダウンタイムは，コンテナ間では 32%，Linux-Windows 間では 60%，Windows 間では 51%，短縮できた．また，FT を無効にした状態と比較しても，最大値及び平均値を短縮できたことがわかる．

5. おわりに

本論文では，これまで開発してきたアタックレジリエントサーバの実装を Bash から vSphere SDK を活用した Perl に変更することにより，最大で 60%のダウンタイムを短縮した．今後は，Windows Server 2016 に新しく導入されるコンテナ機能により，仮想マシン間をコンテナ間の切り替えに変更し，さらにダウンタイムを短縮する．

参考文献

1) Sano. F, at el., A cyber attack-resilient server using hybrid virtualization , Procedia Computer Science, Vol. 96, pp. 1627-1636, 2016.
2) 佐野，岡本，Idris，畑，石田，フォールトトレランスを有するアタックレジリエントサーバの構築とその性能評価，pp.229-236，Computer Security Symposium (2016)
3) Okamoto. T., SecondDEP: Resilient Computing that Prevents Shellcode Execution in Cyber-Attacks, Procedia Computer Science, Vol. 60, pp. 691-699, 2015.