不正アクセス等再現・模倣実験環境の統合に関する一考察

三輪 信介*

インターネット上では時々刻々と新たな攻撃手法が考案され，実行されているため，それらを再現・模倣し，その影響や対策の有効性を検証できる実験環境の構築は重要である．

しかし，このような実験環境を実現するためにさまざまな手法が存在し，統一された再現・検証に関する基準がないため，何をもって正しい検証とするかを見極めることが困難である．

そこで，本稿では，我々が研究開発してきた複数の不正アクセス等の再現・模倣実験環境を挙げ，これらの相互接続手法や統合運用・管理手法についての議論を通じて，不正アクセス等再現・模倣実験環境の統合のあり方について述べる．

A study of an integration of simulating environments for the Internet security incidents

Shinsuke Miwa†

In the recent trend of attacks on the Internet, an attack either trades on some imperfect mechanism of the Internet, or perverts extensive hosts or sites that have deficient security against attacks. The attack sometimes defeats hosts or sites that have sufficient security against attacks, and sometimes has a considerable effect on the Internet widely.

To against these attacks, new countermeasure technologies must be puzzled out from deal with the whole components of the Internet.

Considerable number of experiment environments to simulate these attacks and their efficients were researched and developed. However, we could not distinguish what evaluations are valid because there are no organized baseline for these simulating environments.

In this paper, we deal with an integration of simulating environments for the Internet security incidents through some experiments to integrate between simulating environments.

1 はじめに

インターネット上での種々の攻撃手法に対し，その影響や対策の有効性を検証する実験環境の構築には，さまざまな手法が用いられており，何をもって正しい検証とするかを見極めることは困難である．

そこで，本稿では，いくつかの実験環境を統合する試みを行い，それを通じて，不正アクセス等の再現・模倣実験環境にとって不可欠な要素やその能力の基準となるものを導き出すことを目指す．

本稿で述べる実験環境とは，コンピューターネットワーク，特にインターネットにおける不正アクセス等の再現実験環境を意味し，各要素間での通信や各要素の挙動を再現・模倣し，それらを計測することで，攻撃の影響や対策の効果などさまざまな結果を得ることを目的としたものである．

また，本稿における統合とは，複数の実験環境が相互に影響し合い，かつ，統一的に管理・運用することが可能な状態を指す．

*独立行政法人 通信総合研究所 情報通信部門 非常時通信グループ

†Emergency Communications Group, Information and Network Systems Division, Communications Research Laboratory

2 実験環境の分類

本章では，まず，実験環境の類別を関連研究を参照しながら行う．

2.1 実験環境

インターネットに関する実験を行うために，さまざまな実験環境を構築する試みがなされてきた．本節では，それらを取り上げる．

2.1.1 ネットワークシミュレータ

代表的なネットワークシミュレータとして，NS[1]がある．NSは，Tcl言語によってノードとリンクの特性や動作を記述することで，主にIPネットワークをシミュレートすることができる．既定の動作以外に関しては，C++ 言語で記述することで，独自に追加することが可能である．

IPネットワークに関して，ルータやホストはノードとして高度に抽象化されているため，詳細な挙動に関しては，記述することもシミュレートすることもできない．

2.1.2 ネットワークエミュレータ

NSE[2, 3]は，シミュレータであるNSと実際のIPネットワークとをつなげ，IPネットワークのエミュレーションを実現する．

NSを元にしているため，ルータやホストはNSE内部では高度に抽象化されているため，NS同様に詳細な挙動に関しては，模倣できない．

2.1.3 実機による実験環境

実機による実験環境は，実際に利用されているツールを利用できることや，実装レベルでの問題点を含めて実験できるなど，再現の精度が優れているという特徴がある．

しかし，各要素を構成するノード数はあらかじめ用意された実機の数によって制限される．また，構成に関する柔軟性も低い．

2.1.4 ハイブリッド環境

Emulab[4]は，分散システムとネットワークに関する統合実験環境である．豊富な実験管理機能を持つのが特徴である．

実機とエミュレータの両方を利用可能な複合実験環境で，独自のディスクイメージ切替機構を持つ実PCノードとNSEを利用するエミュレータノード，それらをつなぐ物理ネットワーク接続から構成されている．

物理ネットワークは，ソフトウェアによって変更可能なパッチパネルによって結線を変更でき，VLANなどと合わせて，柔軟にさまざまなネットワークを構成することが可能である．

また，複数の実験環境をRON[5]によってインターネットを介して接続し，より大規模な実験を可能にしている．

Emulabでは，NSEを利用しているため，エミュレータノードでは通信に関する挙動は再現できるが，OSやソフトウェアの実装レベルでの挙動を再現できない．よって，正確な挙動を把握したい要素に関しては，実PCノードに配置する必要があるため，実PCノードの数がそのまま実験規模の制約になる点は，実機で構成した場合と同じである．ただし，通信に関する挙動だけが必要な要素をエミュレータノードに配置することで，規模の制約を緩和できる．

2.1.5 部分的再現・模倣

実験環境に対して，一方的に入力を与える装置や出力を受け取る装置も，部分的に再現・模倣を行う実験環境の一種と考えることができる．例えば，トラフィックジェネレータやネットワークアナライザなどがこれにあたる．

こういった装置は，実験を効率的に行うために有用である．例えば，攻撃者側の要素については詳細な挙動が必要なく，攻撃によってもたらされる通信のみが必要な場合や，どのような通信がもたらされるかを計測できれば良く，それを受ける被害者側の挙動については必要が無いといった場合には，それらの要素の再現・模倣は省略可能である．

よって，このような場合に，詳細な再現・模倣を行うような実験環境を使う必要は無いため，その準備から管理，運用に至る工数を省くことができる．

3 相互接続

複数の実験環境を統合するためには，互いに影響を与えられるようにする何らかの接続が必要である．本章では，実験環境の相互接続について述べる．

本稿においては，実験環境内の対象要素（以降ノードと呼ぶ）について，ノード自身の状態変化などの内部挙動と通信による外部への影響を実験の対象とし，その他の挙動や影響は対象外として，再現・模倣や計測を必要としない．

そのため，相互接続においては，実験環境間でノード同士の通信による影響をどのように伝播するかが重要である．

3.1 物理的接続

まず，相互接続としてそれぞれの実験環境間で通信を行えるようにする物理的接続が考えられる．物理的接続を果たせば，相互接続が実現されるように感じられるが，実際にはそうではない．

例えば，シミュレータを実行している計算機と実機による実験環境を物理的に接続したとしても，実験環境同士が相互に影響を及ぼすことができるわけではないからである．

逆に，物理的に接続を果たしていなくとも，相互に影響を与えることは可能である．例えば，シミュレータに他の実験環境で得られた挙動を入力すれば，シミュレータ内部の挙動に影響を与えることが可能だからである．

このように，物理的接続は実験環境の相互接続に対する十分な条件でも必須の条件でもない．

3.2 論理的接続

次に，相互接続としてそれぞれの実験環境で何らかの影響を伝達する論理的接続が考えられる．シミュレータへ他の実験環境で得られた挙動を入力するなどがこれにあたる．

通信を物理的な通信で実現する実機による実験環境やハイブリッド環境（の一部）では，論理的接続は物理的接続の上に成り立っている．よって，このような環境の論理的接続には，物理的な接続が不可欠である．

シミュレータでは，論理的接続はシミュレータソフトウェア同士の接続やプロセス間通信など必ずしも物理的接続を必要としない．

問題は，シミュレータと実機による実験環境を接続するような場合である．シミュレータは，ある種のソフトウェアであり，シミュレータ内のノードの挙動はすべて，そのソフトウェアの実行実体（プロセスなど）に閉じている．

よって，外部に接続するためには，シミュレータから実ネットワークなどへの変換が必要になる．そのための手法として，NSE や N*(NStar)[6] がある．

3.3 遠距離接続

地理的に離れたところにある実験環境同士を物理的に接続する必要がある場合には，いくつかの問題がある．

物理的な回線を確保することが困難であるということが一つである．近年では，安価に高速な回線を用意することができるようになってきたが，それでも長距離の専用回線を用意するのは困難である．

RON などを用いて，インターネットを利用する方法も考えられるが，セキュリティや実験とは無関係の第三者に影響を与えないなど独自の配慮が必要となる．

また，回線が確保されたとしても，その帯域や遅延によって実験環境をまたがる通信は影響を受けることになる．特に，インターネットを介する場合には，経路制御や回線状況による影響があるため，より複雑になると考えられる．

回線状況に関する問題を解決するための消極的な解決方法としては，実験環境をまたぐ通信が回線状況などの影響などを許容できるようなものに限定されるような実験系の設計を行うことが考えられる．

積極的な方法としては，中間に何らかの変換要素を用意し，回線状況による影響をその要素で吸収することが考えられる．

4 相互運用

複数の実験環境を統合するためには，双方を一つの運用主体によって管理，運用できる必要がある．本章では，実験環境の相互運用について述べる．

相互運用においては，それぞれの管理・運用をどのように行うか，相互の管理における違いをどのように吸収するかが重要である．

4.1 遠隔操作

まず，複数の実験環境がある場合には，それぞれを操作できる必要がある．物理的に一つの計算機の中で実現されている場合には，なんら特別なものを必要としないが，多くの場合は，実験環境は物理的にも分離している．そのため，操作にはなんらかの遠隔操作が必要となる．

BSD 系や POSIX 準拠の UNIX システムであれば，リモート Shell やそれに類するもので操作ができるだろうし，X window system も遠隔操作を提供している．その他の OS でも，多くが何らかの遠隔操作手段を提供している．また，VNC[7] は多くの OS で，インターネットを経由した遠隔操作を可能とする．

また，KVM スイッチや遠隔 KVM などを用いることも遠隔操作の選択肢となりうる．[8]

4.2 さまざまな違い

このような遠隔操作環境を用いることで，運用上の操作に関しては，相互に行うことが可能となるが，それだけで相互運用を果たしたとは言えない．

どの程度忠実に再現するかといった再現の正確性や，時間の取り扱い方，何を計測できるかといった計測の能力，制御がどの程度細かくできるのかといった粒度などは，実験環境毎に固有である．

相互運用を行うには，これらの違いを十分に考慮した上でそれぞれの実験環境の利用方法を決定し，場合によっては，これらの違いを吸収する機構を導入するなどが必要となる．

5 手法の検討

第 1 章に示したとおり，本稿において，統合を果たしたと言えるのは，実験環境が相互に影響し合い，かつ，統一的に管理・運用することが可能な状態である．

本章では，統合する上で，いくつかの手法を検討し，その特徴などについて考えてみる．

5.1 相互接続と遠隔操作

論理的な接続を含めた相互接続を行い，相互に遠隔操作を可能とすることで，統合を果たしたということができる．

ただし，この場合には，4.2 節に述べたような違いを吸収できないため，それぞれの実験環境の特性に応じて，どの実験環境にどのノードを配するかなどに十分な注意を行う必要がある．

5.2 変換機構の導入

論理的な接続を含めた相互接続を行い，4.2 節に述べたような違いを吸収するための変換機構を導入した場合，統合を果たしたということができる．

この場合には，変換機構によって違いを吸収するために，実験環境間の違いは意識する必要が無い．しかし，その代わり，それぞれ実験環境の持つ優位性を変換によって失ったり，全体としての能力が低下するなどの可能性がある．

5.3 併合

論理的な接続を含めた相互接続を行い，一つの実験環境の管理方式で全体を管理するように変更した場合，一つの実験環境に併合する形で統合を果たしたと考えることができる．

ハイブリッド環境は，このような形での統合の例である．Emulab では，実際に NSE 環境が実機による実験環境内に併合されている．

併合の場合には，設計段階で十分に考慮されていなければ，実験環境に大幅な変更が必要となると考えられる．

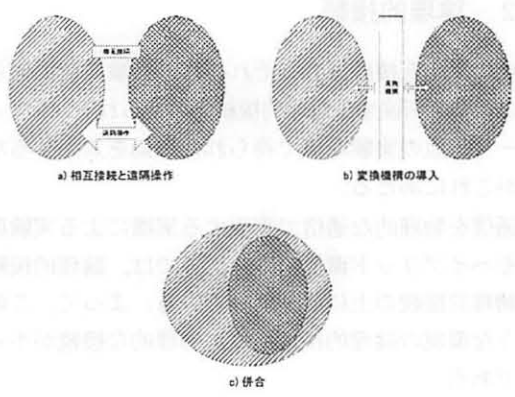

図 1: 統合手法

6 統合実験

実験環境の統合のあり方を探るために，複数の実験環境の相互接続や相互運用の実験を行い，手法を検討する．本章では，我々が研究してきた3つの実験環境について，各環境を概説し，実験について述べる．

6.1 SIOS

大野らによるDDoSシミュレータ[9]は，インターネットセキュリティを対象とした実機による実験環境である．これはSIOS[1]と我々が呼ぶシステムの一部である．

DDoSシミュレータは，100台のPCからなるDDoS攻撃再現部と帯域制御可能なインターネット部，各種のFireWallやIDSを備え，DNS/SMTP/Webなどのサーバを擁する被害者部からなる．(図2)

実際の攻撃ツールを利用して，最大100ノードからのDDoS攻撃を模倣でき，実際の実装を用いて被害者への影響を模倣できる．

また，実験の管理や計測を行うためのエージェントが実装されており，実機による大規模な実験環境でありながら，詳細な実験管理を可能とし，運用の負担を軽減している．

実機で構成されている実験環境の例に漏れず，各部を構成するノード数はあらかじめ用意された実機の数によって制限される．また，構成に関する柔軟性も低い．例えば，攻撃部の一部を被害者部に変更などの柔軟な構成はできず，あらかじめ用意されている攻撃部・インターネット部・被害者部という基本構成に従う必要がある．

6.2 VM Nebula

VM Nebula[10]は，PCエミュレータによる実験環境である．実機による実験環境の再現精度とエミュレータによる実験環境の耐規模性を兼ね備えることを一つの目標とした環境である．

VM Nebulaは，攻撃者模倣用，ネットワーク模倣用（2台），被害者模倣用の計4台のPCエミュレータ実行用のサーバとそれらを物理接続する2台のマルチレイヤスイッチからなる．(図3)

サーバはすべて等価であり，2台のマルチレイヤスイッチはそれぞれすべてのサーバへと接続しているため，実際には各サーバには機能上の違いは無い．そのため，それぞれの役割は変更可能であり，かつ，それぞれの役割への機器の割り当ても任意である．

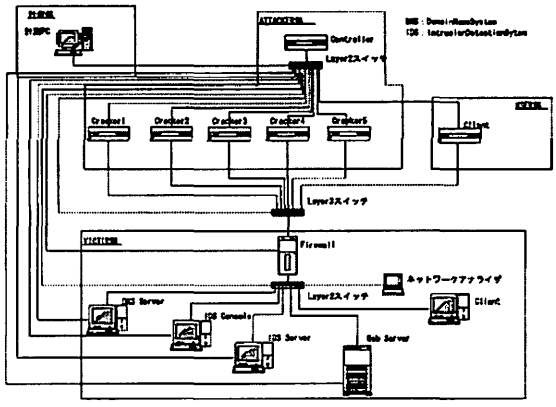

図 2: SIOS DDoS シミュレータ

攻撃者や被害者のPCは，PCエミュレータを用いた仮想PCによって模倣される．FireWallやIDS，各種のサーバからなる被害サイトは，複数台の仮想PCによって模倣される．インターネットは，マルチレイヤスイッチを介してVLANによる接続と帯域制限を行うとともに，ルーティングソフトウェアを実行する仮想PCを仮想ルータとして用いることで模倣する．

実際のOS実装やサーバ実装をそのまま用いることができ，攻撃ツールなどもPC上で動作するものをそのまま利用することができる．

仮想PCの構成やマルチレイヤスイッチの設定などを保存，配布する機能を持っているため，一度構成した実験系を再度構成することや再利用することが容易にできる．

PCエミュレータは1つのサーバで複数実行することができるため，現在4台のサーバで約100台の模倣が可能である．[2]

6.3 不正パケット模倣装置

不正アクセス等に関する影響や対策の検証を実験するためには，被害者やその周辺ネットワークの再現・模倣は必要となるが，攻撃者については詳細な再現・模倣を必要としない場合が多い．そこで，攻撃者からの通信のみを再現・模倣することを目的と

[1] Security Intelligent Operation Studio

[2] PCエミュレータソフトウェアの制限によるもの．PCエミュレータソフトウェアの更新で，256台まで模倣可能であることが確認されている．

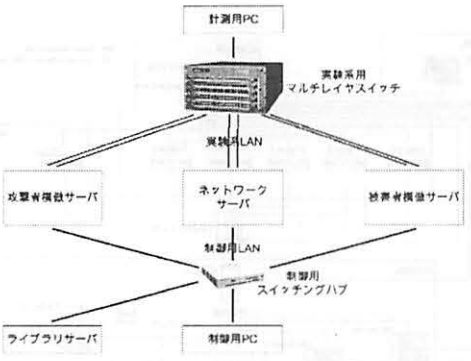

図 3: VM Nebula

した装置が不正パケット模倣装置 [11] である.

不正パケット模倣装置は，トラフィックジェネレータを基にして，任意のパケットを専用の言語で記述された順序とタイミングで発生，送出することができる装置で，DDoS 攻撃などによる大量の通信を再現することを意図している．現在，1Gbps のパケット送出が可能である.

本装置では，被害者からの通信内容に基づいた状態遷移を可能とすることで，状態を有する攻撃を再現することを目標としているが，現在は，外部からの入力を受け付ける機能は無く，記述された内容にしたがってパケットを送出するだけである.

6.4 実験

現在，これらの実験環境を用いた統合実験を計画中であり，近いうちに報じることができる予定である.

7 おわりに

本稿では，インターネットにおける不正アクセス等の再現・模倣実験環境について，そのあり方を，我々が研究してきた実験環境の統合実験を通じて述べた.

参考文献

[1] VINT Project, ⟨URL: `http://www.isi.edu/-nsnam/vint/index.html`⟩.

[2] K. Fall, "Network Emulation in the Vint/NS Simulator", *In proceedings of the 4th IEEE Symposium on Computers and Communications*, 1999.

[3] VINT Project, "Network Emulation with the NS Simulator", ⟨URL: `http://www.isi.edu/nsnam-/ns/ns-emulation.html`⟩.

[4] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar, "An Integrated Experimental Environment for Distributed Systems and Networks (full report)", Netbed Technical Report, May 2002.

[5] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient Overlay Networks", *In Proceedings of 18th Symposium on Operating Systems Principles (SOSP)*, ACM, pages 131–145, Oct. 2001.

[6] 宮地 利幸, 宇夫 陽次郎, 森島 直人, 篠田陽一, "N*(NStar): ns-2 の real external interface の構想", 情報処理学会, マルチメディア通信と分散処理, 103-20, 2001.

[7] Tristan Richardson, Quentin Stafford-Fraser, Kenneth R. Wood, Andy Hopper, "Virtual Network Computing", IEEE Internet Computing, pp33-38, Vol.2 No.1, Jan/Feb 1998.

[8] 大野 浩之, 松本 文子, 山崎 靖博, "高度情報通信危機管理研究施設の構築", 情報処理学会, 分散システム／インターネット運用技術研究会, Apr. 2003.

[9] 大野 浩之, 武智 洋, 永島 秀己, "インターネットの脅威に対抗しうる脆弱性データベースと検証システムの構築", 情報処理学会, DSM シンポジウム 2001, Feb. 2001.

[10] 三輪 信介, 滝澤 修, 大野 浩之, "仮想 PC インターネットセキュリティ実験環境『VM Nebula』の設計と構築", 電子情報通信学会, 2003 年 暗号と情報セキュリティシンポジウム (SCIS2003), 2003.

[11] 三輪 信介, 滝澤 修, 大野 浩之, "トラフィックジェネレータによる DDoS 攻撃の再現", 情報処理学会, マルチメディア通信と分散処理研究会 コンピュータセキュリティ研究会 合同研究会, 2003.