Next Generation Cyber Testing
in a Low Cost Emulation of a Target Network

Deepinder Sidhu and Chuck Burdick
TeleniX Corporation
Tel: 410-772-3275
POC Email: dsidhu@telenix.com

NDIA Test & Evaluation Conference 21-23 July, 2014

Chuck Burdick is an Innovative Decisions, Inc. subcontractor

Realistic, Repeatable, Flexible, Inexpensive, Cyber Testing
• What is the TeleniX Virtual Emulation Environment (VEE) and what does it do?

• How can VEE Support Cyber Testing?

• Questions

Cyber Testing Headlines

• DARPA builds Multi-million National Cyber Range (NCR) with 100s of high-end servers and a dedicated testing facility.

• NCR 5 year support contract awarded sole-source.
Cyber and IA testing on realistic networks is critical, but ranges can be very costly and real networks risky. – But what if you could use actual internet software and protocols of the real network without all the hardware and software costs? • And create actual network configurations in a low-cost virtual emulation – a network clone? • And provide the identical network responses to cyber attack as the real network environment? – And do it running on low-cost computers using actual internet code with bit-level fidelity - essentially duplicating a cyber range on a laptop

Such a realistic network emulation system already exists in the Intelligence Community and is being offered to others – TeleniX Virtual Emulation Environment (VEE)
Clone a network in VEE using:
- Automated Reverse Engineering Techniques
- Actual protocol implementations & network configurations with 100’s of servers, 100K devices
- With complete interchangeability of code between the real and virtual environments

Emulate the network clone in VEE
- Conduct full-fidelity network operations under real-world configurations and operational scenarios
- Produce behaviors that are indistinguishable from the behavior of its real counterpart (confirmed by IC Red Teams)
  - Packet encapsulations, route tables, link bandwidth utilization, ...

VEE on a laptop/server
- Avoid the expense of large-scale hardware and software maintenance/refresh costs, or power, space, & cooling (PSC)
- With minimal personnel support costs
- With rapid reconfigurability and easy portability

VEE uses actual code for all protocols powering the Global Internet

**Internet-in-a-Box**

**VEE Test Advantages**
- Realistic Fidelity
- Repeatability
- Low Cost Test HW
- Fast Reconfiguration
- Full Data Collection

**VEE on a laptop/server**
- Standard Commercial Laptop
  - Contains All Necessary Software
- No External Connections Required
With VEE you can realistically:

**Configure network infrastructure**
- SDH, GigEther, LANs, MANs, WANs, IPv4/IPv6, RIP, OSPF, BGP, LDP, MPLS, DNS, DHCP, Clients, Servers, ...
- SS7, WDM, CDMA, GSM, P2P, VoIP, ...

**Configure network security**
- Firewalls, ACLs, IPSec, IKE/ISAKMP, VPNs, HAIPN, vulnerabilities, malware, NVD, DISA STIGs, ...

**Configure wireless/mobility devices**
- IEEE 802.11, Mobile-IP, MANETs, ...

**Use realistic data sets**
- Sufficient size, proper encapsulations, free from legal issues such as USSID 18
- Have created a 20 million persona database

Reverse engineer networks from data collected on them to see into the network, understand what is in all of the networks and actually be able to defend those networks.
1. Manually – Drag/Drop/Connect
   – Library of pre-config. components
     • Hosts, Routers, switches, ...

2. Automatically Generate Notional Networks
   – # nodes - 50
   – Aver. node degree = 3

3. Reverse Engineer from Network Data Collection
   – Three data feeds:
     • Full capture (top middle rectangle)
     • Router configs (big circle)
     • Netflow (left and right vertical)

Note: Pre-configured components are clones of vendors networking products. They are created based on publically available information about these products.
In addition to automated cloning of actual networks, Testers can build networks that are still in the design stage to evaluate expected network responses and do so with all the fidelity of the implemented network.
Emulating multiple concurrent cyber teams operating on a common cloned network in VEE

Role-Based Multi-Party Web Interface for Simultaneous Red and Blue Teams Operating within the Same Cloned Network

<table>
<thead>
<tr>
<th>Ozone Widgets</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Summary" /></td>
<td>Summary</td>
</tr>
<tr>
<td><img src="image" alt="Infection graphs" /></td>
<td>Infection graphs</td>
</tr>
<tr>
<td><img src="image" alt="Activity graphs" /></td>
<td>Activity graphs</td>
</tr>
<tr>
<td><img src="image" alt="Detailed Logs" /></td>
<td>Detailed Logs</td>
</tr>
<tr>
<td><img src="image" alt="Network Topology" /></td>
<td>Network Topology</td>
</tr>
<tr>
<td><img src="image" alt="Malware Topology" /></td>
<td>Malware Topology</td>
</tr>
<tr>
<td><img src="image" alt="Terrestrial Topology" /></td>
<td>Terrestrial Topology</td>
</tr>
<tr>
<td><img src="image" alt="Report Generation" /></td>
<td>Report Generation</td>
</tr>
<tr>
<td><img src="image" alt="Event Insertion" /></td>
<td>Event Insertion</td>
</tr>
</tbody>
</table>

Graphing Engine – Force Directed Layout Algorithm
## Comparison of Cyber Testing Solutions

### Areas of Concern

<table>
<thead>
<tr>
<th>Basis of Test Environment</th>
<th>Cyber Farm High Fidelity Approaches</th>
<th>VEE High Fidelity Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Custom Hardware/Software</td>
<td>Low-cost laptop to server class multi-core class machine (s)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Expense ($) of Cyber Farm</th>
<th>Millions to tens of Millions</th>
<th>A few Thousands</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Scalability</th>
<th>Limited – adding custom HW/SW upgrade is expensive</th>
<th>Inexpensive – adding commodity machine and/or added functionality is low cost</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Space needed</th>
<th>Dedicated room and rack(s)</th>
<th>Essentially none</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Power/AC to run</th>
<th>Significant for large configurations</th>
<th>Insignificant</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Resources to operate &amp; manage</th>
<th>Dedicated team of administrators and network engineers</th>
<th>User operates and manages his own progression on his own laptop</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Access to Classified Environments</th>
<th>Dedicated SCIF with Electromagnetic Controls surrounding the range</th>
<th>Any SCIF and a small Faraday Cage</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>User control over cyber testing</th>
<th>Limited – may require strict scheduling of times for use</th>
<th>Unlimited – Cyber Testing anywhere and anytime</th>
</tr>
</thead>
</table>

### Problems

- **Non-agile hardware-based solutions**
- **High expense of cyber ranges (100’s servers)**
- **Challenge of rapid reconfiguration of large computer facilities within a Cyber Range**
• Link VEE to networked servers/cyber farms, actual networks, and mobile devices to extend Cyber ranges at minimal cost.
• Link cloned networks & test Systems of Systems at minimal cost.
What Can VEE do for Cyber Testing?

- Provide a documented path to a network’s actual configuration and rapidly build a specific network from real-world software components and configurations.
- Significantly reduce the cost of realistic network tests by performing them on a low-cost computer. Perform many tests simultaneously on separate laptops and demonstrate their repeatability.
- Expand testing by linking cloned networks with real networks and/or cyber farms to create systems of systems, especially for joint and allied interface testing.

New low-cost opportunities to greatly increase the scope and number of high-fidelity network tests conducted.
What Can VEE do for Cyber Testing?

- Rapidly reconfigure by wiping a single machine. Use automated reports to begin analysis almost immediately.
- If planning to reuse a network, save and store the network configuration and rapidly reload it on another laptop.
- House the network test in a normal office environment. Or take a whole network into a SCIF on a laptop.
- Easily swap cloned networks among test organizations.

New efficiencies in cyber facilities, support, test conduct, and post-test analysis.
• Networks Reverse-Engineered from Net Data
• Realistic Responses down to the bit level
• Extensively Instrumented with Network Tools
• Agile, Quickly Reconfigured, Repeatable
• Inexpensive Hardware that’s Easily Expanded
• Interfaces to live systems/devices and actual networks are currently being demonstrated
• Available under license to Government Agencies & Government authorized contractors
Live, unclassified VEE demonstrations are available and can be arranged for Government Agencies & Government authorized contractors.

POCs for VEE users in the IC community can be provided.

Wireshark™ successfully decodes pcap data captured in VEE into packets. Most network tools work as on real nets.
Questions?

POC:
Dr. Deepinder Sidhu
Chief technologist
TeleniX Corporation
dsidhu@telenix.com
410-772-3275

Low-Cost, High Fidelity Cyber Testing Using VEE
Network Cloning

- Clone behavior is indistinguishable from the real network
- Clone requires no validation since it is identical to its real counterpart
- All decisions in clone made by actual code and network state – no randomness
- Clone evolves to actual system
- Clone answers any/all questions about net over its life-cycle
- Virtual host/routers in network clone run complete TCP/IP stack under FreeBSD kernel as in real net
- Clone uses identical code and configurations of a real network
- Clone can be used to diagnose and solve operational problems such as routing
- Clone uses 100% of actual code

Network Modeling

- No mathematical basis for the model to behave like a real system
- Virtually impossible to validate a model-based network
- Many decisions in network model made by calling random numbers
- Models often thrown away after use
- Often build new models to answer new questions
- Model has no OS kernel in model nodes, mimics TCP/IP using small amount of code in nodes, runs as app
- No model has ever become reference implementation of any Internet protocol
- Model “mimics” some limited aspect of a network with small amount of code
- Typically uses <20% code with abstractions