



Emulating an Embedded Firewall

Clifford Neuman, Deepak Dayama,
Arun Viswanathan

Clifford Neuman

Director, USC Center for
Computer Systems Security

<http://clifford.neuman.name>

USC Viterbi
School of Engineering

UNIVERSITY OF SOUTHERN CALIFORNIA
**INFORMATION
SCIENCES
INSTITUTE**

DETER Community
Workshop on Cyber
Security and Test

August 7, 2007

Boston

Modeling embedded firewall

- **Reporting on work to model embedded firewall on the Deter Testbed**
 - This work was done by a graduate student Deepak Dayama, and work to improve the support for others is being done by another student Arun Viswanathan.
- **Work funded by DHS**
 - As a sub-contract to Adventium Labs
- **The work is still in progress**
 - Basic support completed, but alternate representation, and final experiments on policy dissemination still in progress.

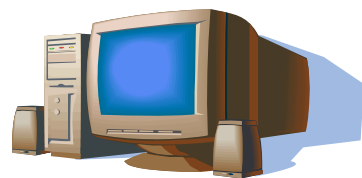
Common network topologies

- **Simplify emulation of common topologies**
 - Host based firewalls common – if not standard today
 - Can be included in standard node images
 - But management would need to be identical
 - Or set up by experimenter on case by case basis
- **Distributed Firewalls**
 - We are starting to see deployments
 - Emulation can still be handled on standard nodes

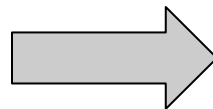
Common network topologies

- **Embedded Firewalls**

- Separate hardware implements firewall function
 - Prevents manipulation by infected host
 - Simplifies host software compatibility
- Emulated initially on DETER as second node



Node A



Node A''



Node A'

Adventium Labs Conversation Manager

- **Simplifies Distributed Firewall Management**
 - Manages policies for distributed embedded firewalls
 - Manager defines conversations
 - Groups of host/ports that can communicate
 - Crypto/IPSec requirements for allowed conversations
 - Manages keys for such conversations
 - All other communication rejected
- **Creates Virtual Networks**
 - Security is managed independent of the “host” nodes.
 - Harder for node “owner” to subvert

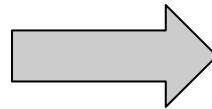
Emulation on DETER

- **We were not provided with the hardware**
 - Emulation of the hardware functions was required.
 - Decided on straightforward emulation of the firewall embedded 3com NIC card as a separate node on DETER for each outfitted experimental node.
 - Required $2N$ nodes to emulate N hosts.
- **Firewall implementation as NetFilter**
 - NIC card used node running Linux
 - Netfilter implementation supported firewall functions.
 - Policy language for conversations converted to iptables directives and groups managed with ipsets.
 - Installed by scripts on updates.

Sample Conversation Manager Output

This is our representative input

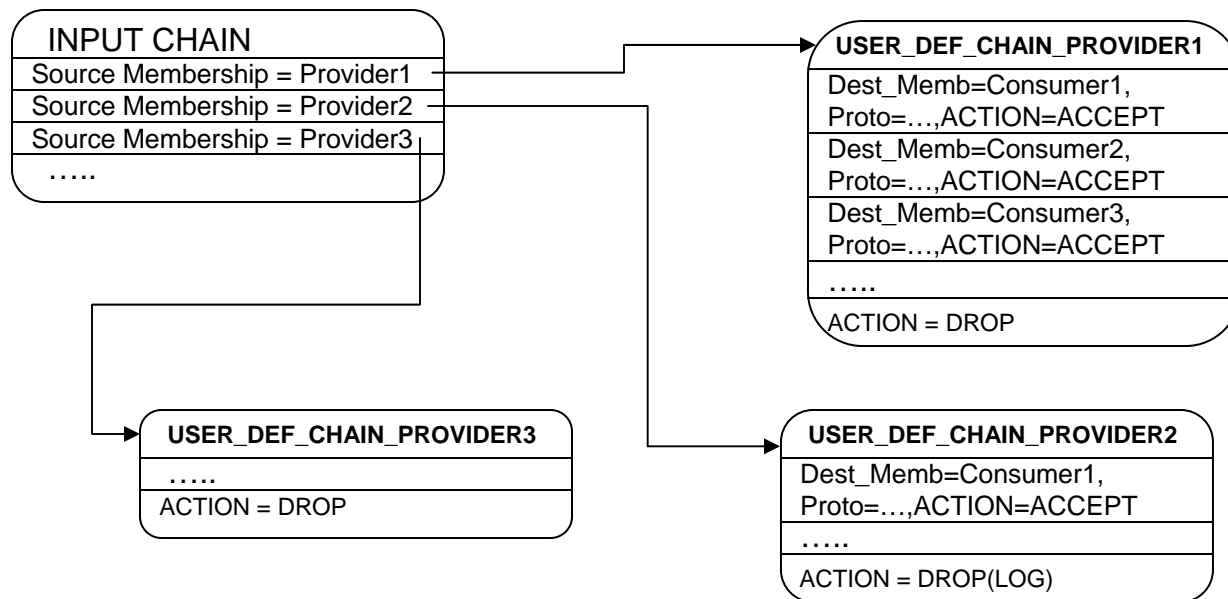
- host1 DNS client host1 host2 NULL
- host1 DNS client host1 host4 NULL
- host1 HTTP client host1 Any IP NULL
- host1 HTTPS client host1 Any IP NULL
- host1 POP3 client host1 host5 NULL
- host1 SMTP client host1 host8 NULL
- host2 DNS server host1 host2 NULL
- host2 DNS server host3 host2 NULL
- host2 DNS server host5 host2 NULL
- host2 DNS server host6 host2 NULL
- host2 DNS server host7 host2 NULL
- host2 DNS server host8 host2 NULL
- host3 DNS client host3 host2 NULL
- host3 DNS client host3 host4 NULL
- host3 HTTP client host3 Any IP NULL
- host3 HTTPS client host3 Any IP NULL
- host3 POP3 client host3 host5 NULL
- host3 SMTP client host3 host8 NUL



- host1 DNS client host1 host2 NULL
- host1 DNS client host1 host4 NULL
- host1 HTTP client host1 Any IP NULL
- host1 HTTPS client host1 Any IP NULL

- Rule applied on
- Port info
- Role (client/server/both)
- Source
- Destination
- Protocol
- Crypto Key if any

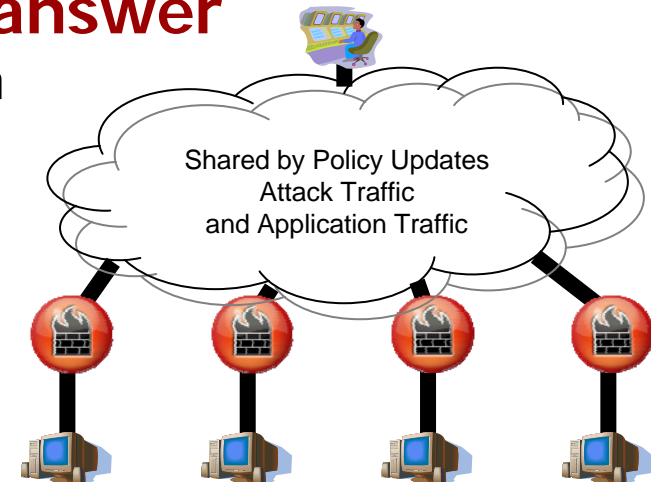
IPtable traversal



Experiments on DETER Testbed

- **What question do we hope to answer**

- Looking at issues of policy distribution within the distributed firewall
 - Push policies to the nodes
 - Pull them from the nodes
 - Hierarchical distribution
- Time to disseminate updates
 - Resilience to denial of service from attacks the policies mitigate.



Issues

- In small topologies differences in approaches might not be noticeable.
- For small topologies effects of different firewall implementation and time to translate policies will dominate results.

Future Emulation

- **When is a separate node needed to emulate the firewall**
 - Can we avoid the need for twice as many experiment nodes as emulated hosts?
 - Consider when the extra host is needed in practice
 - If the software on the user node does not support the firewall functions and policy dissemination.
 - When experimenting with malicious code that in the real network could disable software firewalls.
- **Solutions**
 - Modules for several host images supporting the firewall functions and installation of the EFW policies.
 - Determine impact of malicious code based on characteristics already considered when deciding on containment regimen.
- **For the experimenter**
 - Ideally they should indicate if they want to emulate an EFW, HBFW, DFW, and tools like sser and workbench emulate in most efficient manner consistent with experiment.

For More Information

For updates and related information

- <http://clifford.neuman.name/publications/2007/200708-usecdw-emulating-embedded-firewall/>
- <http://www.isi.edu/deter>
- <http://www.deterlab.net>
- <http://www.emulab.net>
- <http://www.adventiumlabs.org>

This material is based on research that was supported by funding from the United States National Science Foundation (NSF) and the United States Department of Homeland Security (DHS) under contract numbers ANI-0335298 (DETER) and CNS-0454381 (DECCOR) and by the United States Air Force and the Department of Homeland Security HSARPA as a subcontract to Adventium Labs under contract number FA8750-05-C-0144. Opinions, findings, conclusions and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation, the United States Air Force, the Department of Homeland Security or Adventium Labs.