Visualizing Firewall Configurations Using Created Voids

Shaun P. Morrissey
Georges Grinstein
Proof of Concept: Visualize a Firewall Configuration

- Firewall rulesets are large, order-dependent, legacies
  - Current tools are text-editors
- Goal: Apply visualization to management & comprehension
  - Problem: no known work
- Solution: Proof of Concept
  - Build Interactive Ruleset Editor
  - Create Graphics Pipeline for Firewall Configuration
  - Try lossless visualization to start
  - Find usable dataset
Rules & Ruleset Semantics

• Firewall rule as sextuple in two parts
  • Predicate – range or interval (upper and lower limit) in five dimensions
    • Dimensions: Source Address, Source Port, Protocol Number, Destination Port, Destination Address
  • Action: Accept or Deny
• Packets processed one at a time by
  • Testing against predicates in rule order
  • First match determines action ("rule firing")
• Predicate overlap and order-dependence can create problems
Calculate Acceptance Volume

- Guttman algorithm
- Constructive Solid Geometry
  - Integer lattice
  - 5 dimensions - Penteracts
  - Axis-aligned – intervals only
- Modifications
  - Add Provenance (rules)
  - Add Created Voids
  - Convex solid decomposition
Penteract Constructive Solid Geometry (3D analogue)

- Add Provenance of rules
  - List of rules
  - Connected to editor
- Modify Guttman: B - A
  - Normal: discard $B \cap A$
  - Created Void: retain & label with joint provenance
- Creates visualizable artifact

Top face of rule A box (red) has been opened to expose $A \cap B$
Use Convex Solid Decomposition

- Simple Data Structure
  - Only penteracts required
- Calculation Complexity
  - 371,293 types of overlap
  - CSD allows one dimension at a time, five cuts, 13 cases
  - Cost: longer list
- Convex penteract can be visualized easily
  - Parallel Set Enclosure

Rule A: red volumes
Rule B: green volumes
\( B \cap A \): blue volume
1-D cuts
Set operations as disposition rules for convex solid decomposition lists

<table>
<thead>
<tr>
<th>Operation</th>
<th>A – B</th>
<th>A ∩ B</th>
<th>B – A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Union</td>
<td>Keep</td>
<td>Keep</td>
<td>Keep</td>
</tr>
<tr>
<td>Intersection</td>
<td>Discard</td>
<td>Keep</td>
<td>Discard</td>
</tr>
<tr>
<td>Set Difference</td>
<td>Keep</td>
<td>Discard</td>
<td>Discard</td>
</tr>
<tr>
<td>Void Difference</td>
<td>Keep</td>
<td>Re-label &amp; Keep</td>
<td>Discard</td>
</tr>
<tr>
<td>No.</td>
<td>Source</td>
<td>SourcePort</td>
<td>Protocol</td>
</tr>
<tr>
<td>-----</td>
<td>-----------------</td>
<td>------------</td>
<td>----------</td>
</tr>
<tr>
<td>1</td>
<td>Yuan R1source</td>
<td>10.1.1.0/25</td>
<td>all ports</td>
</tr>
<tr>
<td></td>
<td>0:65535</td>
<td>0:65535</td>
<td>TCP 6</td>
</tr>
<tr>
<td>2</td>
<td>All</td>
<td>255.255.255.2...</td>
<td>all ports</td>
</tr>
<tr>
<td></td>
<td>0:65535</td>
<td>0:65535</td>
<td>all ports</td>
</tr>
<tr>
<td>3</td>
<td>Yuan R3source</td>
<td>10.1.1.128/25</td>
<td>all ports</td>
</tr>
<tr>
<td></td>
<td>0:65535</td>
<td>0:65535</td>
<td>TCP 6</td>
</tr>
<tr>
<td>4</td>
<td>Yuan R4&amp;7 so...</td>
<td>172.16.1.0/24</td>
<td>all ports</td>
</tr>
<tr>
<td></td>
<td>172.16.1.0/24</td>
<td>0:65535</td>
<td>all ports</td>
</tr>
<tr>
<td>5</td>
<td>Yuan R5&amp;6 so...</td>
<td>10.1.1.0/24</td>
<td>all ports</td>
</tr>
<tr>
<td></td>
<td>10.1.1.0/24</td>
<td>0:65535</td>
<td>TCP 6</td>
</tr>
<tr>
<td>6</td>
<td>Yuan R5&amp;6 so...</td>
<td>10.1.1.0/24</td>
<td>all ports</td>
</tr>
<tr>
<td></td>
<td>10.1.1.0/24</td>
<td>0:65535</td>
<td>all ports</td>
</tr>
<tr>
<td>7</td>
<td>Yuan R4&amp;7 so...</td>
<td>172.16.1.0/24</td>
<td>all ports</td>
</tr>
<tr>
<td></td>
<td>172.16.1.0/24</td>
<td>0:65535</td>
<td>all ports</td>
</tr>
<tr>
<td>8</td>
<td>All</td>
<td>255.255.255.2...</td>
<td>all ports</td>
</tr>
<tr>
<td></td>
<td>0:65535</td>
<td>0:255</td>
<td>all ports</td>
</tr>
</tbody>
</table>
Lossless Parallel Coordinate View

Rule allowing a Class A address access to an HTTP server
Finding Firewall Data for Analysis, Comparison, & Publication

- Requests for firewall configuration examples
  - Occasional examples provided for internal use
  - Uniform **Absolute** Denial of Permission to Expose
  - One use case released
- Text Firewall Correctness tools appear in the literature
  - Al-Shaer and Hamed 2003, Firewall Policy Analyzer
  - Yuan, et al. 2006, FIREMAN (modeling and analysis)
- Firewall Anomalies – predicate overlaps
  - Al-Shaer & Hamed – defined all possible anomalies
Compact Created Example Includes All Anomalies: Al-Shaer & Hamed

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Source</th>
<th>Destination</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Address</td>
<td>Port</td>
<td>Address</td>
</tr>
<tr>
<td>1</td>
<td>tcp</td>
<td>140.192.37.20</td>
<td>any</td>
</tr>
<tr>
<td>2</td>
<td>tcp</td>
<td>140.192.37.*</td>
<td>any</td>
</tr>
<tr>
<td>3</td>
<td>tcp</td>
<td>***<em>.</em></td>
<td>any</td>
</tr>
<tr>
<td>4</td>
<td>tcp</td>
<td>140.192.37.*</td>
<td>any</td>
</tr>
<tr>
<td>5</td>
<td>tcp</td>
<td>140.192.37.30</td>
<td>any</td>
</tr>
<tr>
<td>6</td>
<td>tcp</td>
<td>140.192.37.*</td>
<td>any</td>
</tr>
<tr>
<td>7</td>
<td>tcp</td>
<td>140.192.37.*</td>
<td>any</td>
</tr>
<tr>
<td>8</td>
<td>tcp</td>
<td>***<em>.</em></td>
<td>any</td>
</tr>
<tr>
<td>9</td>
<td>udp</td>
<td>140.192.37.*</td>
<td>any</td>
</tr>
<tr>
<td>10</td>
<td>udp</td>
<td>***<em>.</em></td>
<td>any</td>
</tr>
<tr>
<td>11</td>
<td>udp</td>
<td>***<em>.</em></td>
<td>any</td>
</tr>
</tbody>
</table>

Ruleset created to exercise Firewall Policy Analyzer
Use Case: Email server collision with legacy service protection rule

DNS and SMTP
Created Voids. Six total penteracts with two voids created by the interaction of the 32760 deny rule’s cleaving. The two voids are contained within the remainder of the DNS and SMTP accept rules.
All Anomalies Combined: PC Visual of Al-Shaer & Hamed 2003

23 penteracts are presented clearly highlighting the need for interactive data zooming and multiple views.
Contributions & Directions

- Configuration visualization is feasible
- Created Voids useful for interaction and visualization
- Occlusion quickly becomes issue
- PC view does not capture containment of one volume or void within another, the set-subset relationship

Development Directions
- Data windowing controls
  - Specialized two-dimensional controls
- Alternate Visualization
Flow Picture Mockup: Pipe-Through-the-Wall Metaphor

FP Representation. The plane in the left of the object represents the source address and source port axes. The destination address and destination port are similarly set up as a two-dimensional plane. The remaining value, the protocol number, is treated as a single axis.
Backups
Anomaly Examples in 2-D

Shadowing

Generalization

Correlation

Redundancy
What’s out there?

And the research literature on firewall visualization was simply “None” until 2007.