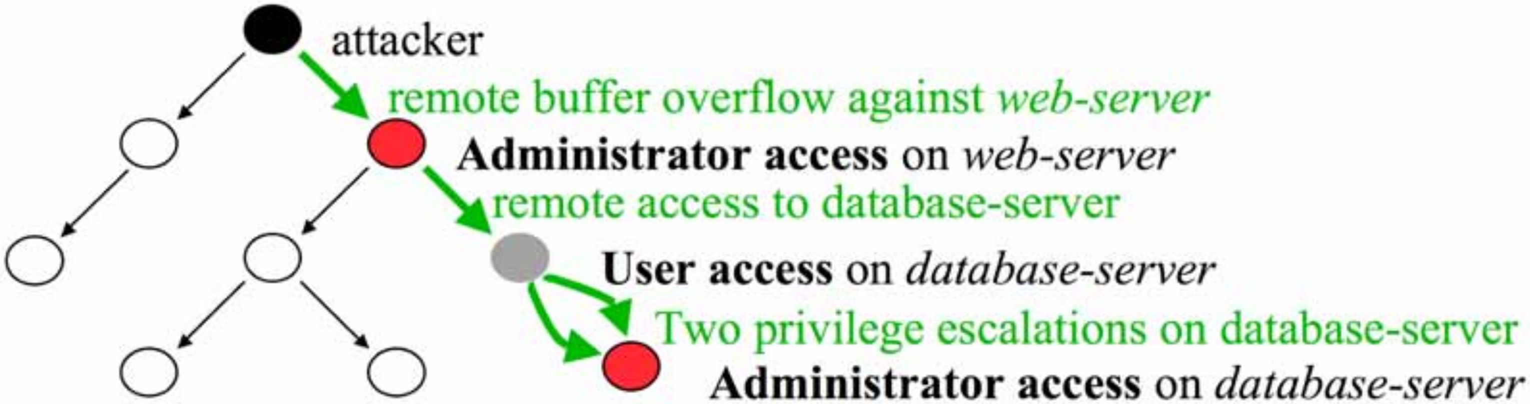
An Interactive Attack Graph Cascade and Reachability Display

Leevar Williams, Richard Lippmann, Kyle Ingols

29 October 2007

Introduction

- Attack graphs are useful tools in assessing network security
  - Provide a way to model attacker behavior
  - Reveal critical weaknesses in network



attacker

remote buffer overflow against *web-server*

Administrator access on *web-server*

remote access to database-server

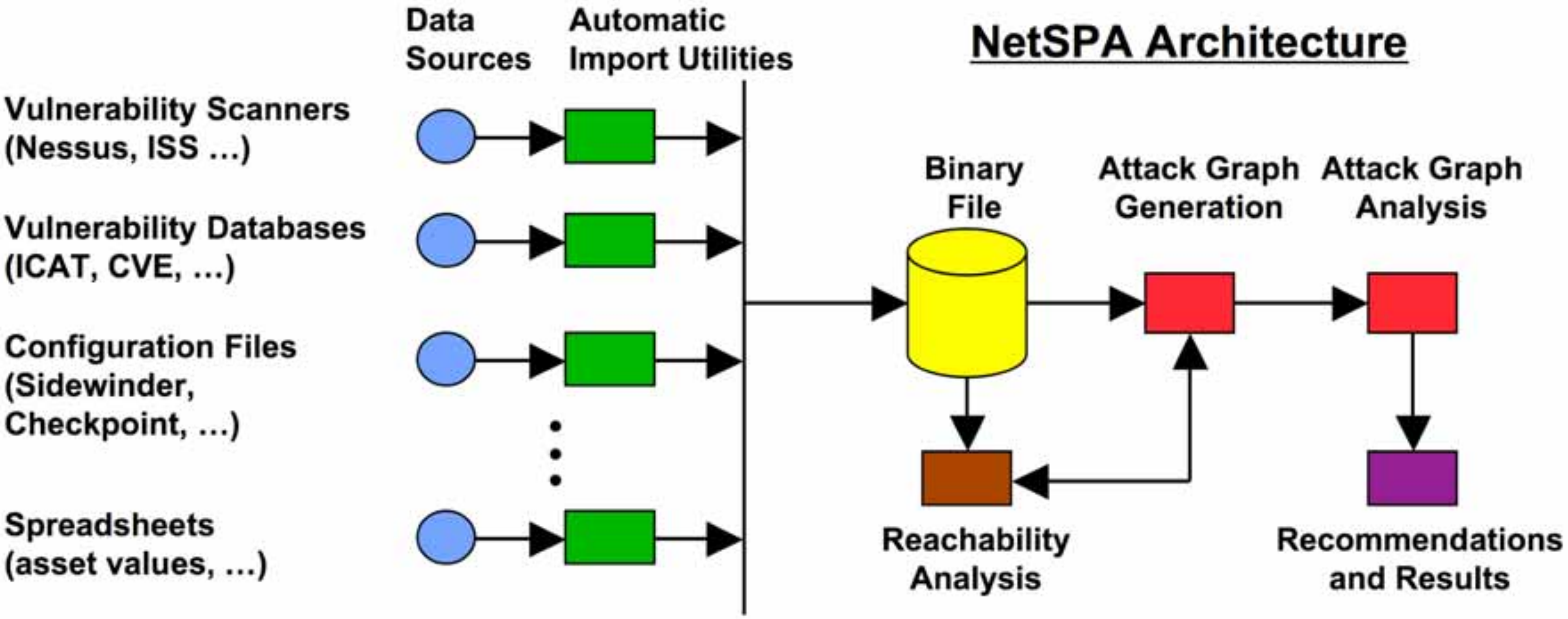
User access on *database-server*

Two privilege escalations on database-server

Administrator access on *database-server*

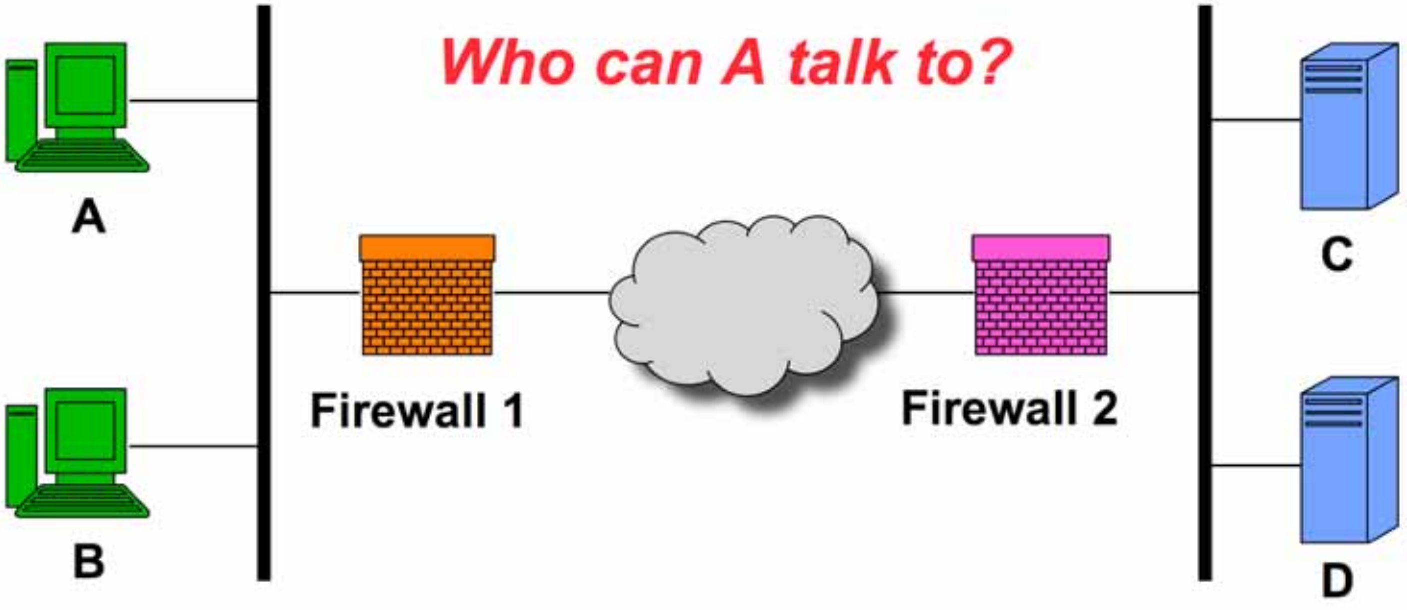
- Constructed by calculating how attacker can use multiple vulnerabilities to progress through network

NetSPA System


Data Sources
Automatic Import Utilities
NetSPA Architecture
Vulnerability Scanners (Nessus, ISS …)
Vulnerability Databases (ICAT, CVE, …)
Configuration Files (Sidewinder, Checkpoint, …)
Spreadsheets (asset values, …)
Binary File
Attack Graph Generation
Attack Graph Analysis
Reachability Analysis
Recommendations and Results

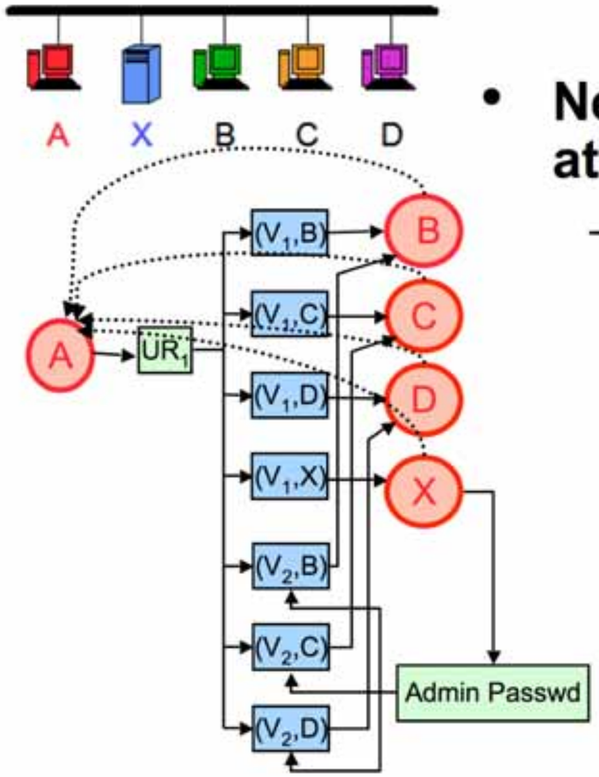
- NetSPA (NETwork Security and Planning Architecture) tool represents one approach to attack graph generation
- Imports data from vulnerability scanners, firewall rulesets, and vulnerability databases
- Computes reachability and attack graph, and produces set of recommendations to protect vulnerable hosts

Host-to-Host Reachability



- Reachability calculations involve finding all target hosts / ports that can be reached from each source host
- Determined by reading in and analyzing firewall rules

Multiple-Prerequisite Attack Graph



- **NetSPA produces multiple-prerequisite (MP) attack graphs**
  - **Consists of three node types**

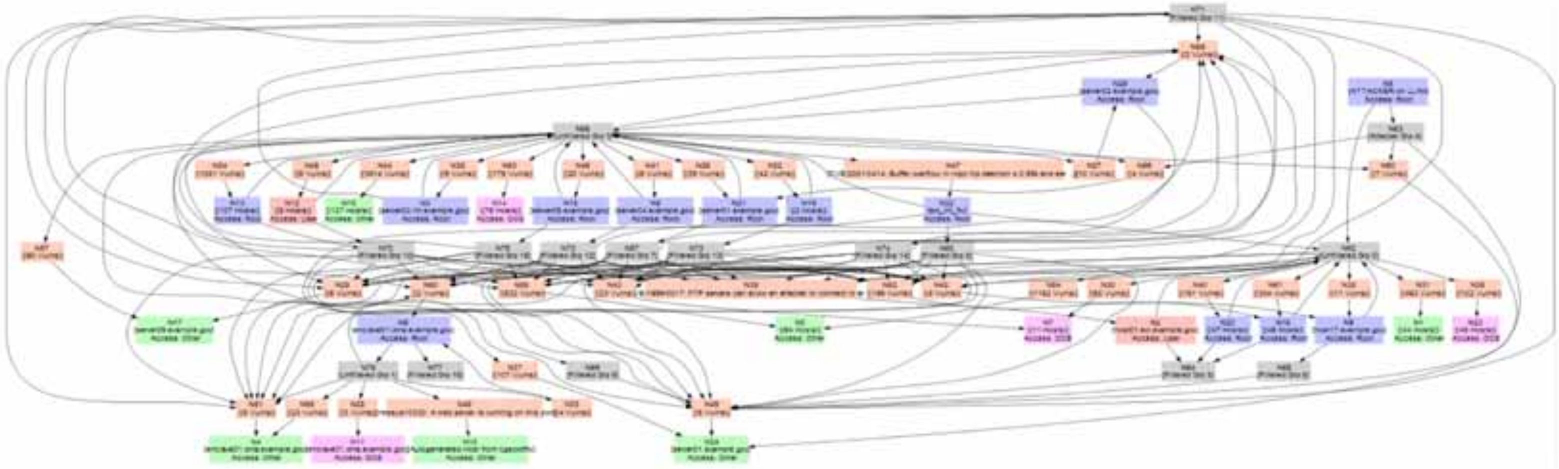
    State nodes represent attacker's level of access on a particular host (i.e. root, user, DoS, other)

    Prerequisite nodes represent reachability or some sort of credential needed to exploit a vulnerability

    Vulnerability Instance nodes represent a particular vulnerability on a specific host port

- **Graph simplified by collapsing together state nodes with identical reachability, trust relationships, and compromise level**

Limitations of Previous Attack Graph Visualization Approaches

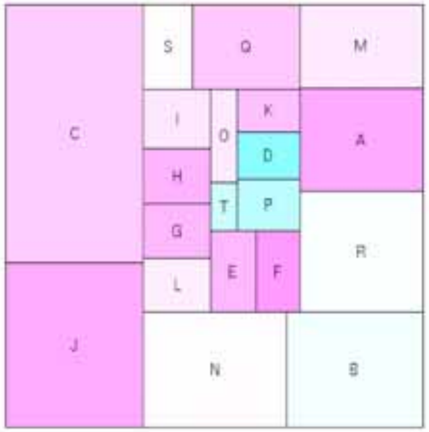


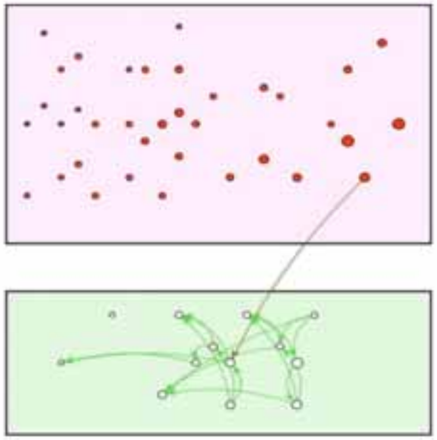
- **Complete graphs are difficult to visually navigate**
  - **Grow unacceptably large and complex with many nodes and crisscrossing edges**
  - **Can be simplified by node grouping and hierarchies, but often remain difficult to interpret**
- **Displays are not intuitive for network administrators**
  - **Positioning of nodes does not correspond to physical layout of network**
  - **Host-to-host reachability usually not displayed**

Design Goals for New Approach

- **Simplified display will facilitate understanding of and interaction with attack graphs by:**
  - **Highlighting critical attack steps where an attacker may jump between subnets or compromise valuable groups of hosts**
  - **Partitioning hosts into groups representing fully connected domains (e.g. subnets, VLANs)**
  - **Illustrating general reachability between hosts to aid in understanding progression of attacker**
  - **Allowing direct interaction to manage links and rearrange groups into desired topological positions**

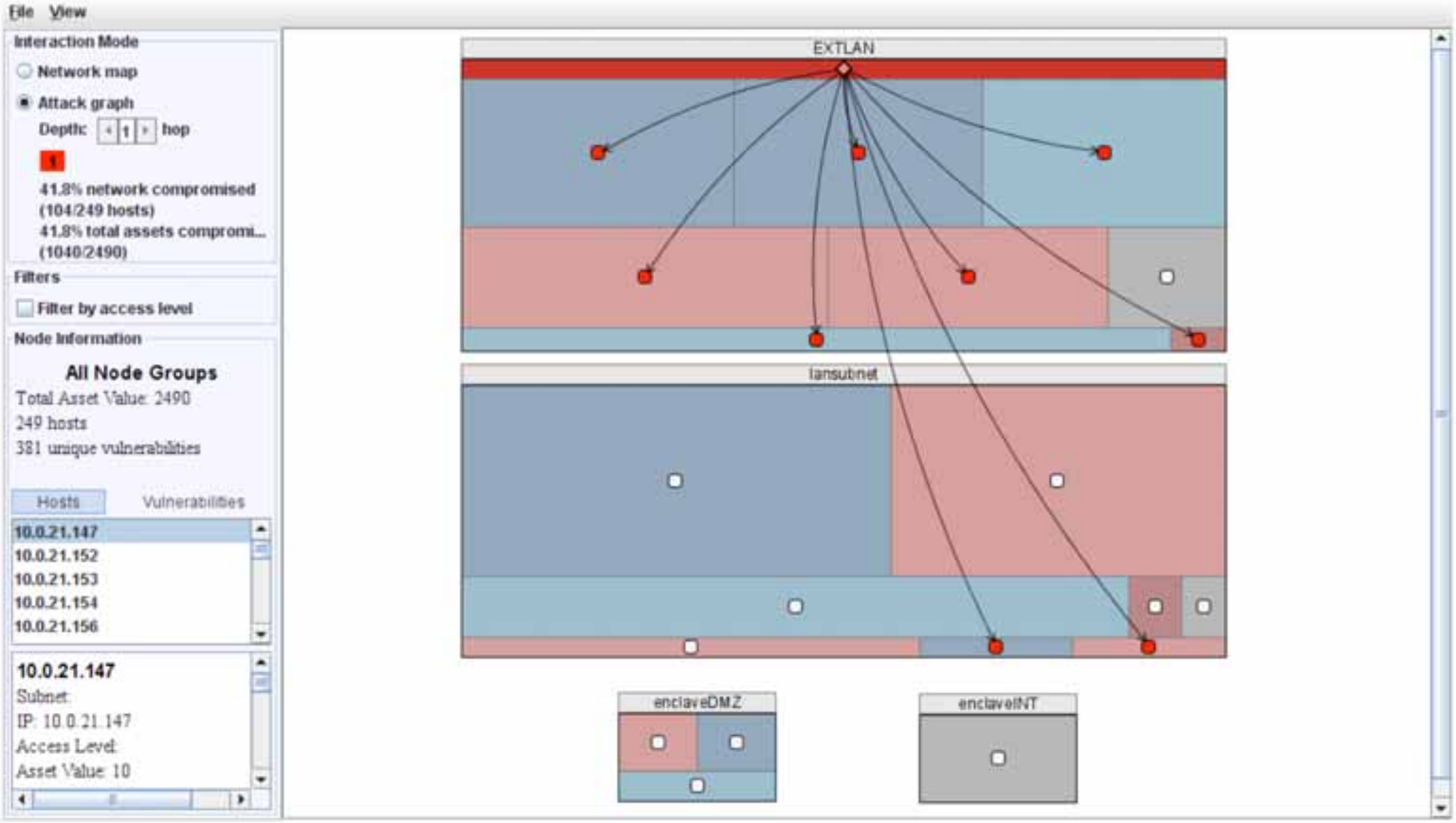
Display Overview





- **Display combines two main visualization techniques**
  - **Treemaps (Johnson and Shneiderman, 1991)**
    - **2D space-filling approach**
    - **Divides display area into set of nested rectangles**
    - **Rectangle sizes proportional to some attribute of data**
  - **Network Visualization by Semantic Substrates (Shneiderman and Aris, 2006)**
    - **Layout based on user-defined *semantic substrates***
    - **Nodes placed in non-overlapping regions according to some attribute**
    - **Offers interactive control of node and link visibility**

Display Overview



Interaction Mode

○ Network map

● Attack graph

Depth: 1 hop

1

41.8% network compromised (104/249 hosts)

41.8% total assets compromi... (1040/2490)

Filters

□ Filter by access level

Node Information

All Node Groups

Total Asset Value: 2490

249 hosts

381 unique vulnerabilities

10.0.21.147
10.0.21.152
10.0.21.153
10.0.21.154
10.0.21.156

10.0.21.147

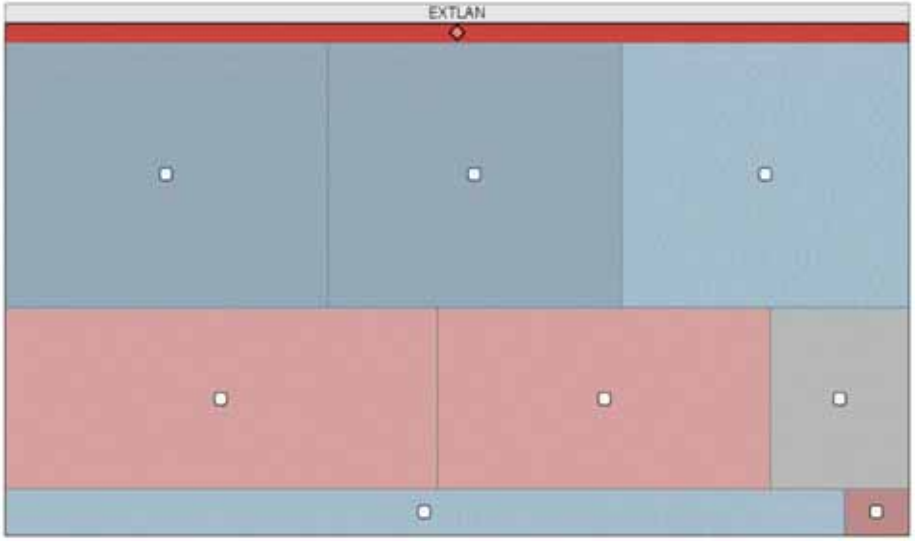
Subnet:

IP: 10.0.21.147

Access Level:

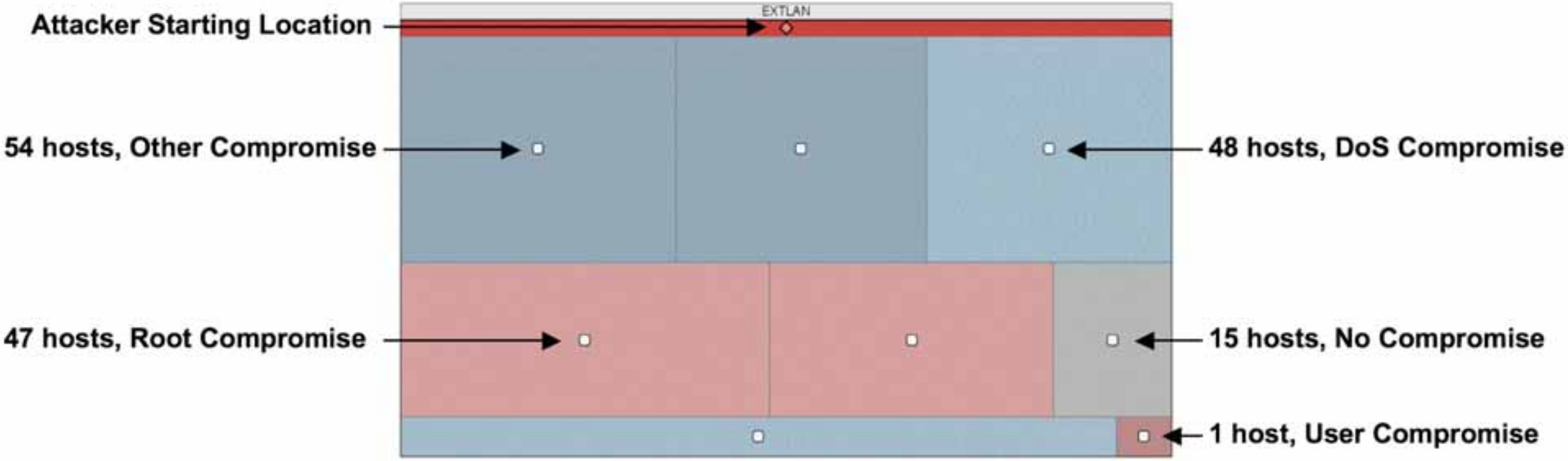
Asset Value: 10

Network/Graph Visualization



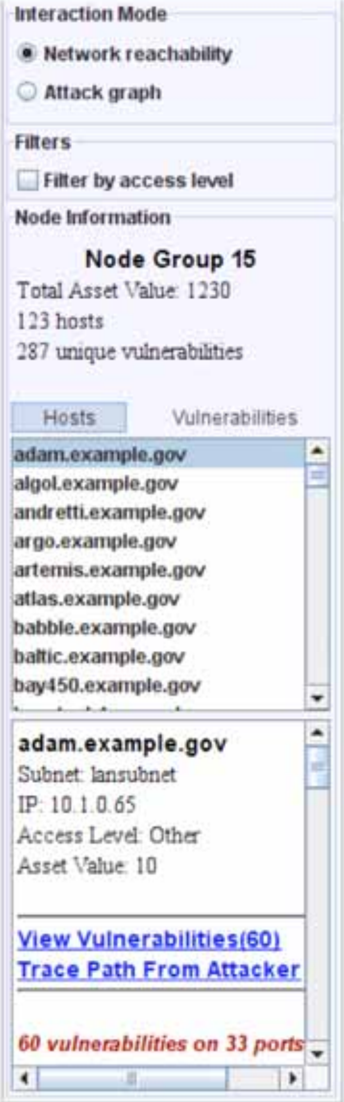
- Only collapsed state nodes from MP graph drawn in display
- Nodes grouped by subnet; placed in labeled rectangle
- Each node placed in nested rectangle and laid out according to strip treemap algorithm
- Nested groups proportional in size to number of represented hosts nodes and colored by compromise level

Network/Graph Visualization


EXTLAN
Attacker Starting Location
54 hosts, Other Compromise
48 hosts, DoS Compromise
47 hosts, Root Compromise
15 hosts, No Compromise
1 host, User Compromise

- Only collapsed state nodes from MP graph drawn in display
- Nodes grouped by subnet; placed in labeled rectangle
- Each node placed in nested rectangle and laid out according to strip treemap algorithm
- Nested groups proportional in size to number of represented hosts nodes and colored by compromise level

User Interface



Interaction Mode

● Network reachability

○ Attack graph

Filters

□ Filter by access level

Node Information

Node Group 15

Total Asset Value: 1230

123 hosts

287 unique vulnerabilities

adam.example.gov
algol.example.gov
andretti.example.gov
argo.example.gov
artemis.example.gov
atlas.example.gov
babble.example.gov
baltic.example.gov
bay450.example.gov

adam.example.gov

Subnet: lansubnet

IP: 10.1.0.65

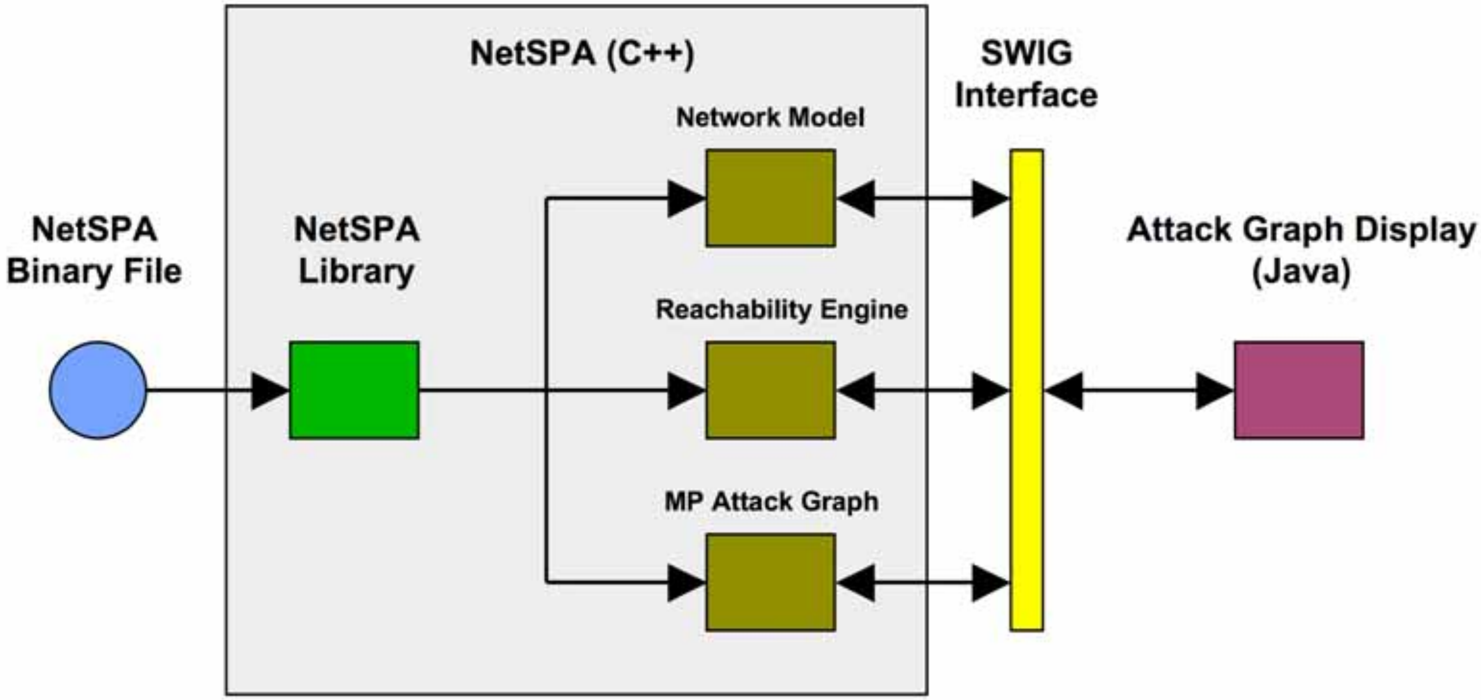
Access Level: Other

Asset Value: 10

60 vulnerabilities on 33 ports

- **Two modes of interaction**
  - **Network reachability mode**
  - **Attack graph mode**
- **Side panel exposes controls for selecting modes and filtering node groups**
- **Includes information about selected node group**
- **Lists represented hosts and vulnerabilities and individually provides data for each**
- **Tooltip-like context menus provide subset of information and allow control of displayed links**

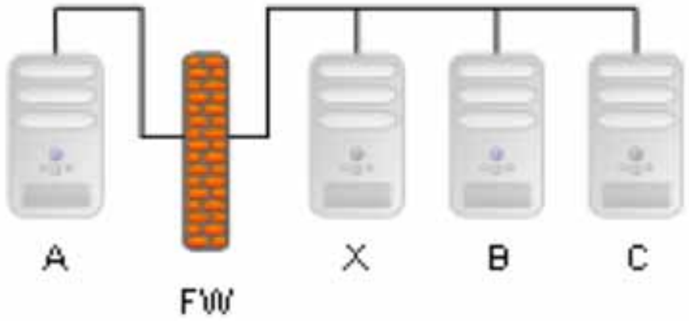
Implementation Details

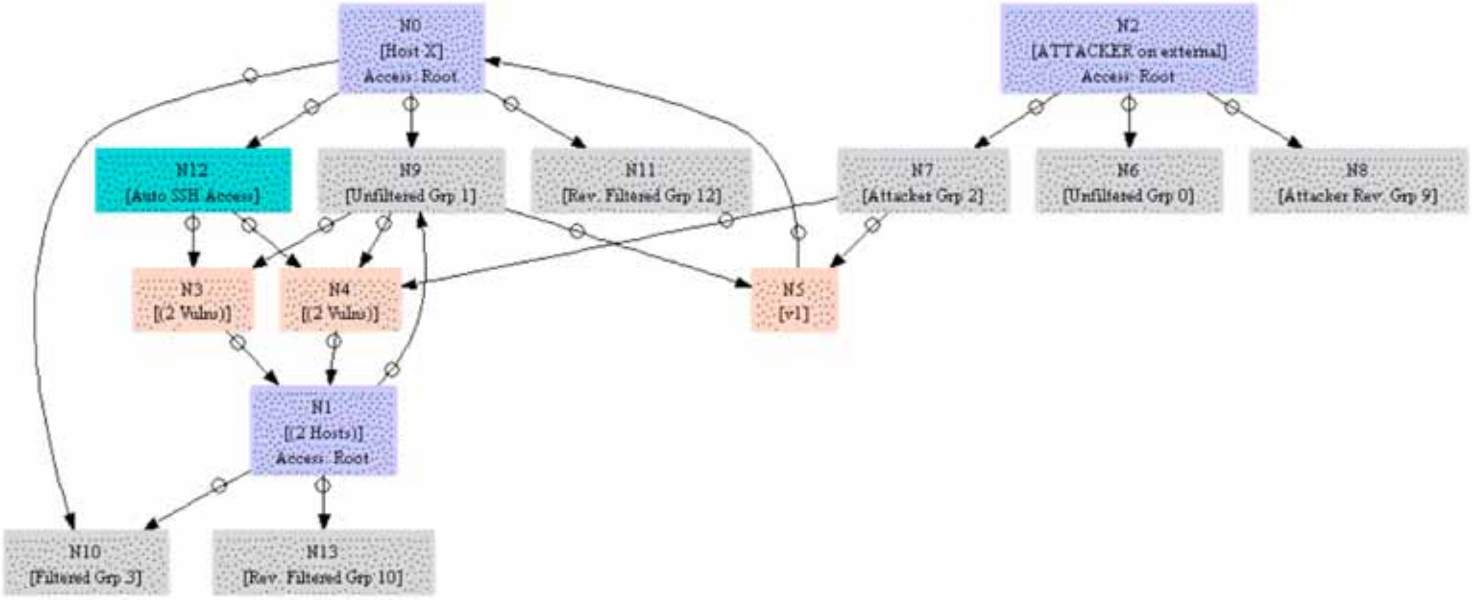


- **Display implemented in Java using Swing and other third-party libraries**
- **NetSPA library loads binary network model and produces C++ objects for data access**
- **Java and C++ code communicate using SWIG Toolkit**

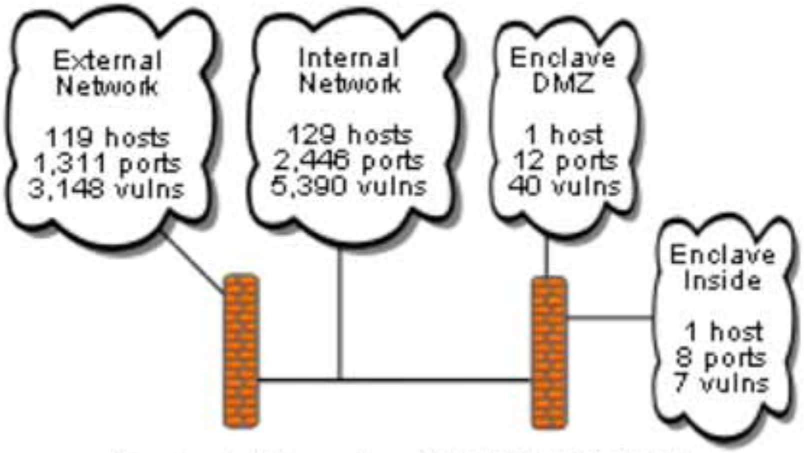
DEMO

Demo – Example Network





Demo – Field Test Network



External Network

119 hosts
1,311 ports
3,148 vulns

Internal Network

129 hosts
2,446 ports
5,390 vulns

Enclave DMZ

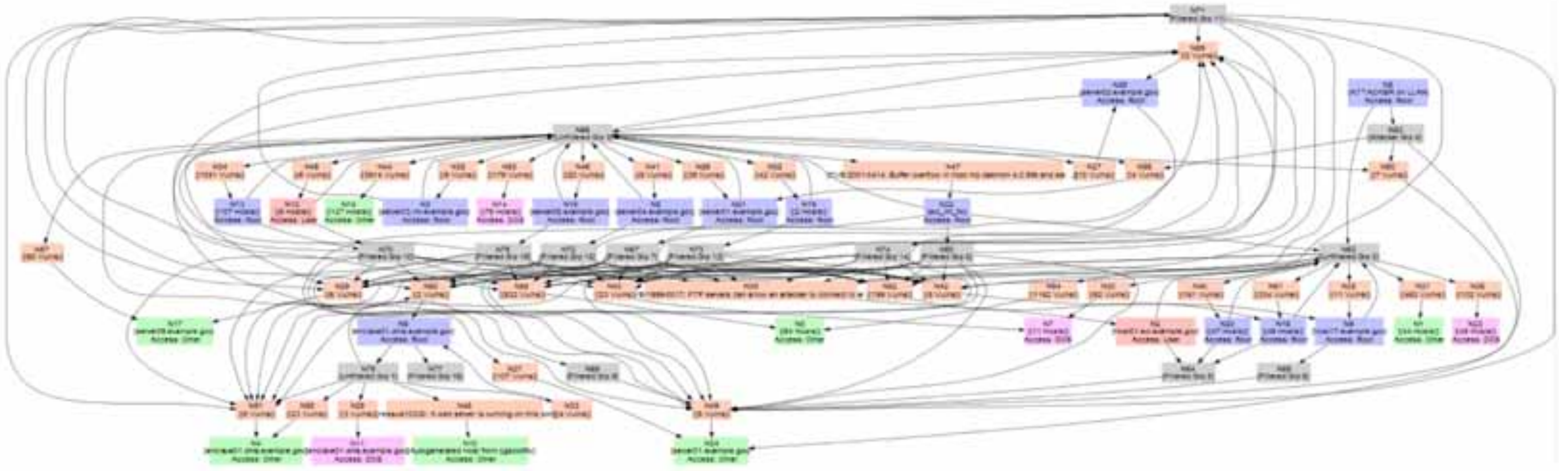
1 host
12 ports
40 vulns

Enclave Inside

1 host
8 ports
7 vulns

Checkpoint Firewall
75 filtering rules
5 NAT rules

Sidewinder Firewall
83 filtering rules
37 NAT rules



Summary

- Developed a new combined attack graph and reachability display
- Hosts in each subnet displayed within treemap rectangles
  - Rectangles positioned manually to reflect physical or logical topology
  - Hosts automatically grouped by reachability and level of compromise
- Incremental interactive display shows critical attacker hops into new subnets and what vulnerabilities allow this
- Can also be used to explore reachability within network
- Rapid interaction made possible by using C++ engine and Java display

QUESTIONS