



DETER/EMIST

DDoS Defense Experimental Methodology

Overview

Stephen Schwab

June 15, 2006



Team Participants

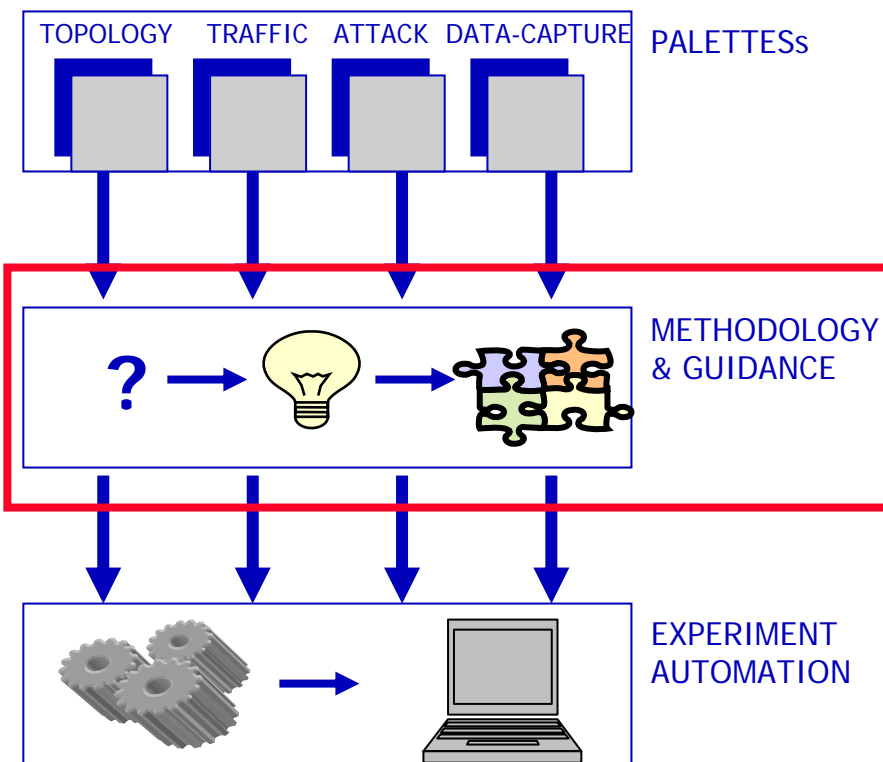
- **SPARTA:**
- **DETER**
 - Steve Schwab, Ron Ostrenga, Richard Edell, Dan Sterne
- **EMIST DDoS**
 - Steve Schwab, Brett Wilson, Calvin Ko, Ron Ostrenga, Roshan Thomas, Alefiya Hussain, Dan Sterne
- **Collaborators:**
- **EMIST DDoS**
 - Sonia Fahmy (Purdue), Roman Chertov (Purdue)
- **DDoS Benchmarking**
 - Jelena Merkovic (University of Delaware)



Objectives for EMIST

- **Create reusable library of test technology for conducting realistic, rigorous, reproducible, impartial tests**
 - For assessing attack impact and defense effectiveness
 - Test data, test configurations, analysis software, and experiment automation tools
- **Provide usage examples and methodological guidance**
 - Recommendations for selecting (or developing) tests and interpreting results
 - Test cases and results, possibly including benchmarks
- **Facilitate testing of prototypes during development and commercial products during evaluation**

Security Experiment Methodology



- **Experimenter's select from a palette of predefined elements: Topology, Background and Attack Traffic, and Data Capture and Instrumentation**
- **Our Methodology frames standard, systematic questions that guide an experimenter in selecting and combining the right elements**
- **Experiment Automation increases repeatability and efficiency by integrating the process to the DETER testbed environment**



Near Term Objectives

- Define canonical form for DDoS experiment
- Develop examples to illustrate, explore canonical form
 - DDoS defense research prototypes, e.g., FloodWatch, Cossack
 - DDoS defense commercial products, e.g., ManHunt, Sentivist
- Create/enhance tools to automate experiments and analyze results
- Focus on how to conduct experiments. Objective is not to determine whether a particular DDoS defense technology “works”
- Identify/clarify requirements for DETER facilities

- Additional DDoS defenses: D-Ward, CloudShield Appliance Entropy detector, Peer-to-peer Overlay, etc.



Canonical DDoS Experiment

- **DDoS experimentation involves a vast multidimensional space of experimental variables.**
- **Canonical experiment form intended to organize the experimental space and facilitate navigation through it.**
- **Canonical experiment consists of:**
 - 1. Attack Mechanisms**
 - 2. Background Traffic**
 - 3. Network Topology**
 - 4. Defense Mechanisms**
 - 5. Measurements and Metrics**
 - 6. Network Services Infrastructure**
 - 7. Risk**

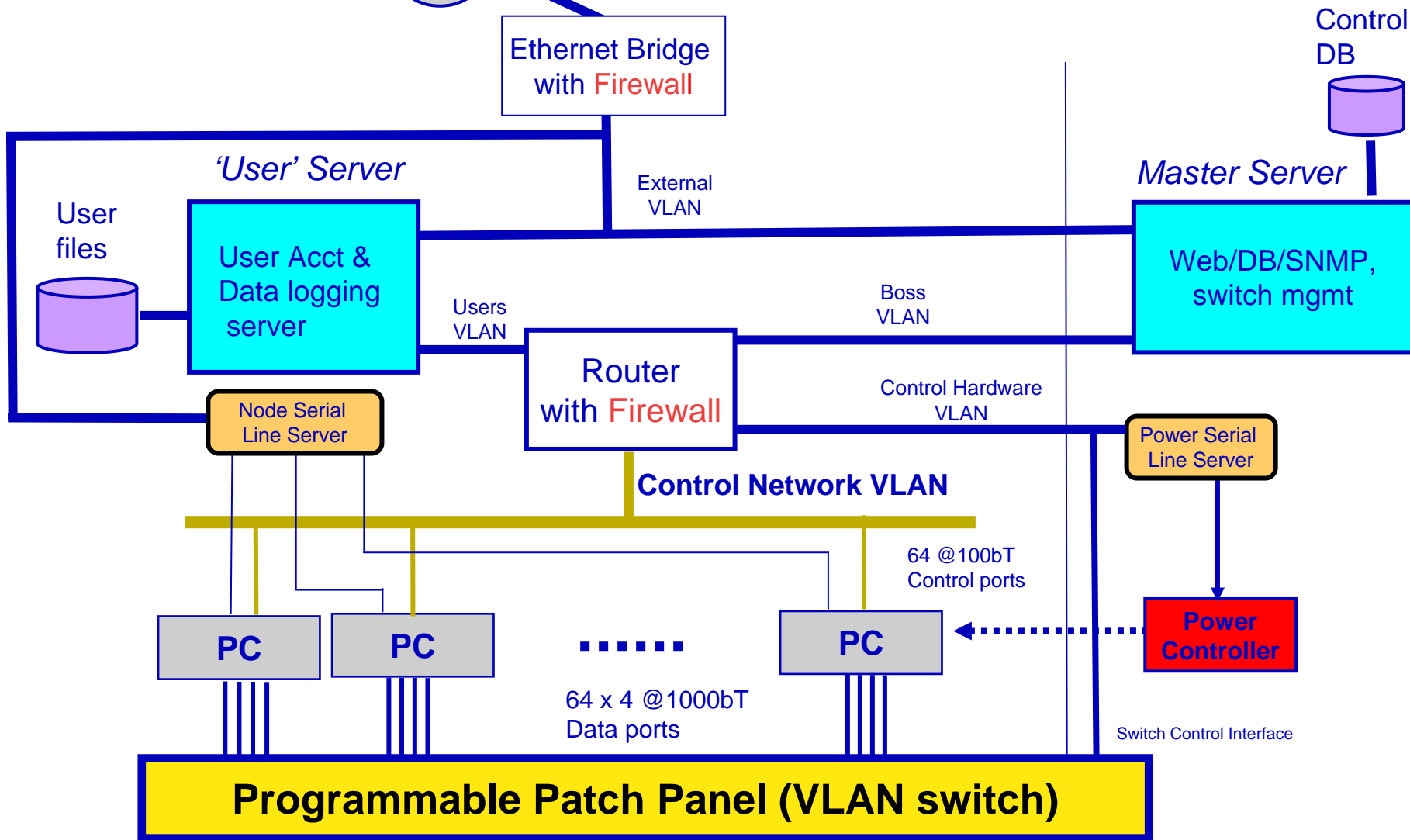


What is an experiment?

- **Philosophy of Science**
 - Kuhn's Paradigm Shift
 - Differing Scientific Theory's are Incommensurable
- **Proponents of competing paradigms have different ideas about the importance of solving various scientific problems, and about the standards that a solution should satisfy.**
- **The vocabulary and problem-solving methods that the paradigms use can be different: the proponents of competing paradigms utilize a different conceptual network.**
- **The proponents of different paradigms see the world in a different way because of their scientific training and prior experience in research.**



DETER Testbed Cluster Architecture





SPARTA DDoS Experiment September 2005

Background Traffic:

REPLAY | NTCG | HARPOON

HIGH FIDELITY TRAFFIC

Topology:

BUILDING-BLOCKS |

JUNIPER ROUTER CORE

*REALISTIC CONNECTIVITY AND
SCALE-DOWN*

Attack Traffic:

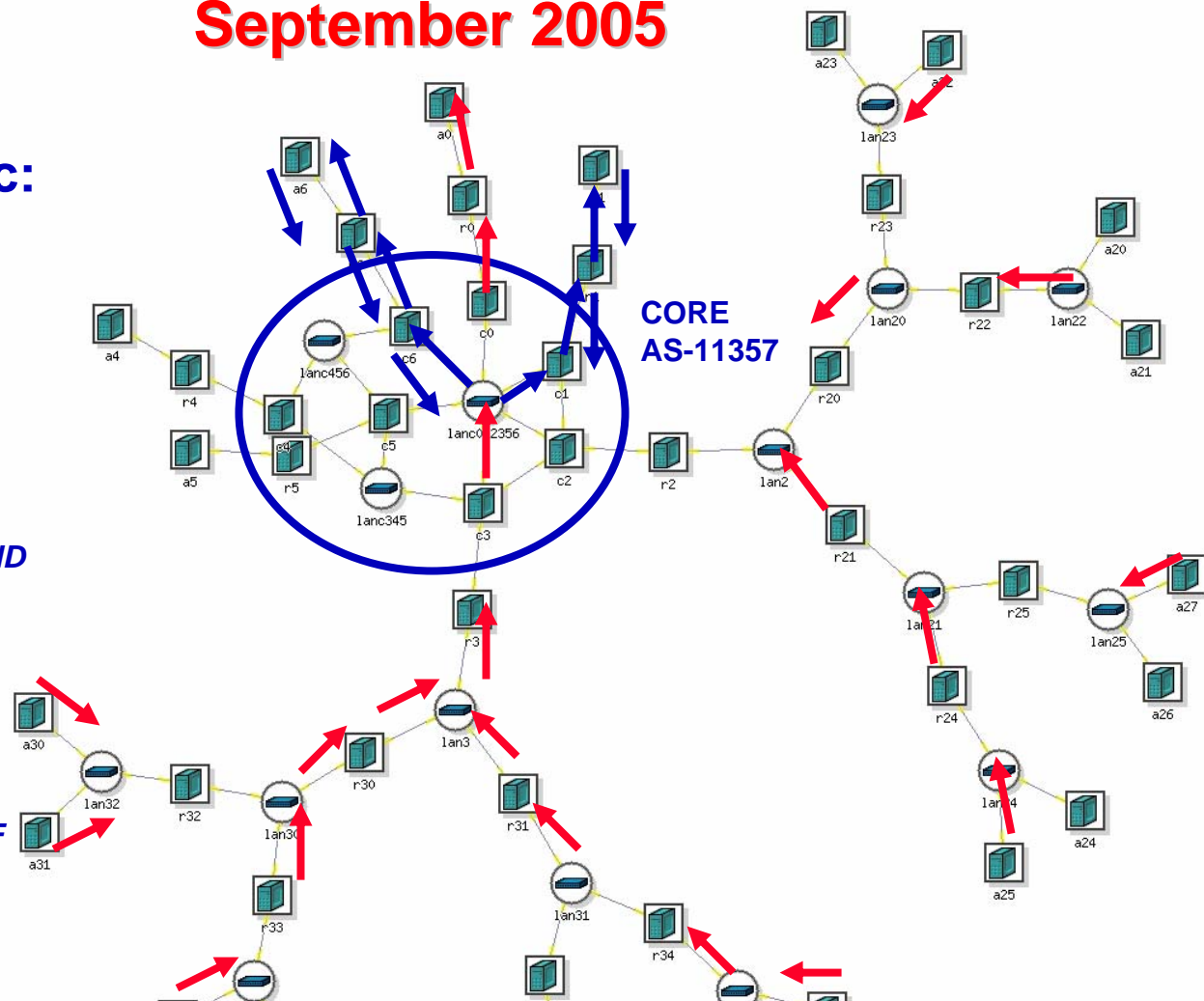
DETER-INTEGRATED ATTACK
SCRIPTING

*AUTOMATION OF VARIETY OF
SCENARIOS UNDER STUDY*

Instrumentation:

PACKET AND HOST STATISTICS
CAPTURE | SPECTRAL ANALYSIS
| METRICS CALCULATION |
INTEGRATED VISUALIZATION

*TOOLBOX FOR RIGOROUS
INVESTIGATION OF RESULTS*



**This unique “view” of
DETER’s resources is part
of our “vocabulary”**

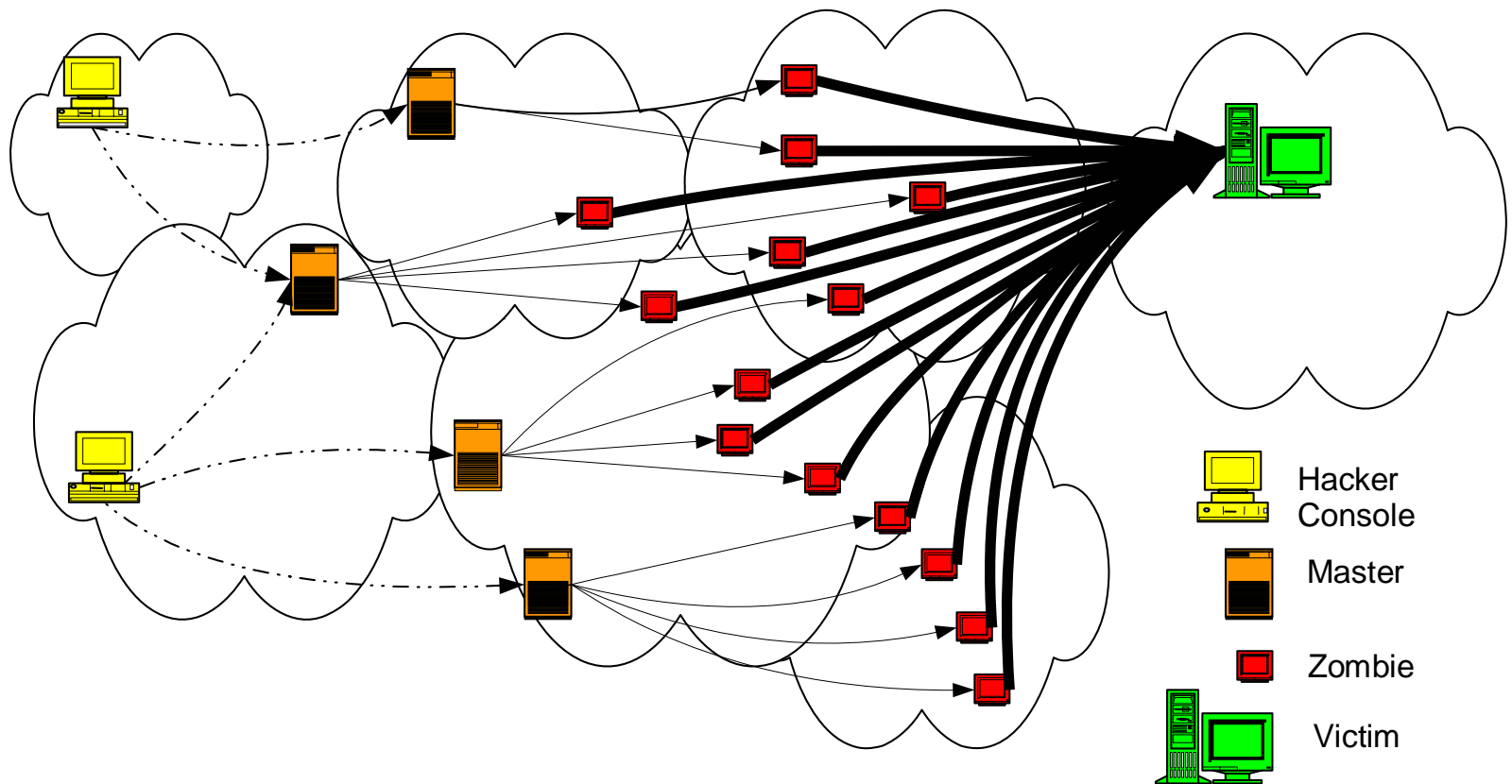
ATTACK TRAFFIC
BACKGROUND TRAFFIC



1. Attack Mechanisms

- **DDoS attack tactics:**
 - Flood/congest network links
 - Exhaust computational resources of hosts or infrastructure
 - Cause crashes
 - Combinations of the above
- **DDoS attack tools utilize large numbers of penetrated bystander systems (“zombies”) using a control hierarchy**
- **Worms may be used to penetrate and acquire zombies rapidly (e.g., MyDoom).**
- **BotNet projects beginning to use DETER**

Typical DDoS Toolkit Architecture





1. Attack Mechanisms (cont)

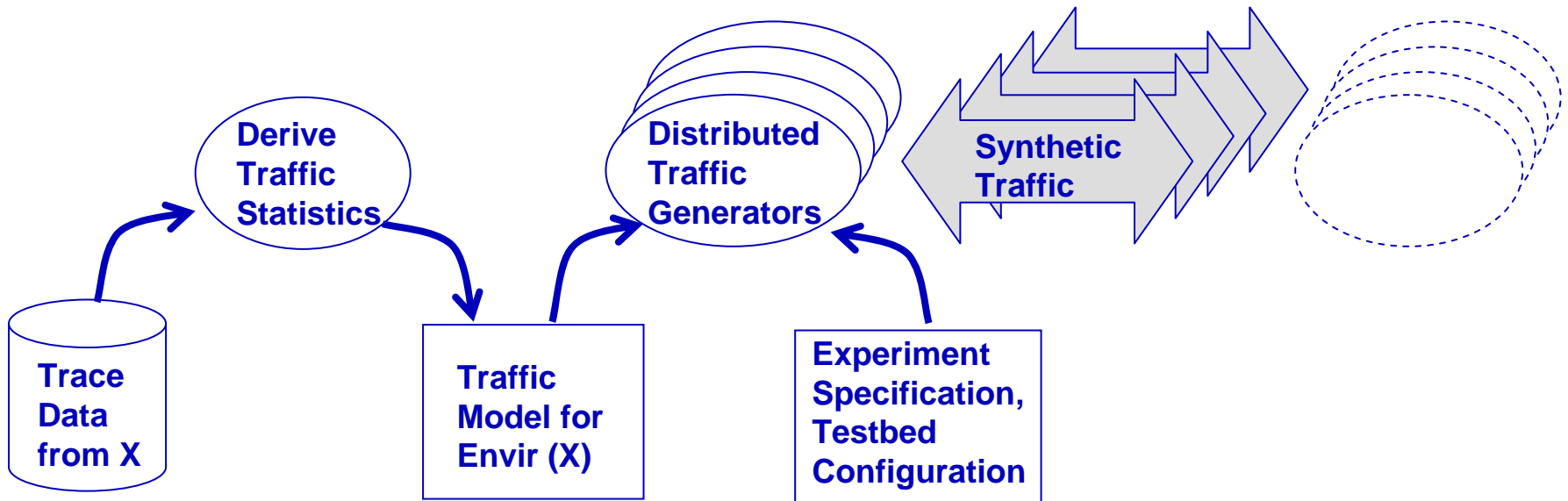
- **Attack traffic generated by most DDoS attack hacker tools is crude**
 - e.g., uniform randomization of spoofed source addresses, fixed packet size, single protocol
 - Sophistication is primarily in disguising control traffic to zombies
- **Future attack traffic will be more sophisticated**
 - Packets that “blend in” statistically with background
 - Pulsating, time-varying, location-varying (whack-a-mole)
 - Adaptive, real-time control
 - Targeted at (or leverages impact) on Internet infrastructure
 - Massively-replicated full-fledged normal connections (MyDoom)
- **Custom attack traffic generators often needed for experimentation**
- **Several attack agents or tools used by different experimenters**



2. Background Traffic

- **Without realistic background traffic, intrusion detection/response is trivial**
- **Creating realistic, representative traffic is complex**
 - Replaying packet traces provides statistical fidelity but sacrifices the dynamics and adaptivity of live traffic (esp. TCP congestion control)
 - » Replays must be remapped to testbed topology/addresses
 - » Fidelity depends on trace source and test environment being similar, e.g., both are web server farms
 - » Traces for some environments difficult to obtain; require sanitization, esp. if payload is required
 - Generating synthetic traffic can provide dynamics, but may lack statistical fidelity. Need to drive synthesis with accurate statistical models of different environments.

Generating Synthetic Traffic





2. Background Traffic (cont)

- Test environment for DDoS should provide both replays and synthetic traffic
- Although traffic generators and benchmarks exist, e.g., WebStone, SURGE, NetSpec, *constantly evolving* models are necessary
- Current traffic generators: Harpoon, UDP VoIP model

V. Paxson and S. Floyd. "Why We Don't Know How to Simulate the Internet." In Proceedings of the 1997 Winter Simulation Conference, Dec. 1997



3. Network Topology

- **Experiment topology should approximate *each* DDoS defense's intended deployment environment, e.g.,**
 - Core: FloodWatch
 - Edge: Cossack
 - Enterprise: ManHunt, Sentivist
- **How to model topology of Internet with limited nodes is a challenging research problem**
 - Precisely capturing transit and stub networks through power-law and small-world phenomena is non-trivial
 - Link delay and bandwidth information is also important



3. Network Topology

- **For DDoS experimentation**
 - Create library of synthetic topology building blocks that abstractly represent commonly occurring patterns:
 - » Examples: Fan-in tree, stub network, AS peering relationships, backbone, etc.
 - » Leverage topology generators
 - Router-level topology generators, e.g., Calvert/Zegura96, Jin/Bestavros02
 - AS-level generators, e.g., Towsley02, Govindan02, Jamin01/02, Matta01
 - » Leverage real Internet topology data selectively where more fidelity is needed (e.g., U of Oregon RouteViews, NLANR data, CAIDA Internet tomography)
 - Assemble synthetic topologies together to approximate intended deployment environment
- **Topology generation and configuration: Purdue tools (Rocketfuel-to-ns), SPARTA tools**



4. Defense Mechanisms - Objectives

- **Remove/reduce attack traffic**
 - Detect
 - » Typically via volume, statistical signature or anomaly
 - » Easier near victim; harder upstream (traffic dilution, need instrumentation at many locations)
 - Discriminate benign vs. attack traffic
 - » Packet marking to determine origin
 - » Special authorization for improved service
 - » Statistical characteristics
 - Rate-limit or discard attack traffic; more effective upstream
 - Determine or shutdown source of attack traffic; impeded by address spoofing
- **Improve host/infrastructure resource management algorithms (e.g., syn cookies)**



4. Defense Mechanisms

- **DDoS defense mechanism variations**
 - Objectives: detect, mitigate, or traceback
 - Positioning: enterprise, edge, core
 - Degree of communication/coordination required



5. Measurements and Metrics

- **Many metrics are potentially applicable to DDoS experiments**
 - Metrics for measuring attack impact
 - » Impacts occur at multiple levels, e.g., network, application, host
 - » Impacts can cause chain reactions, i.e., primary, secondary effects
 - Metrics for measuring defense effectiveness
 - » For detectors: true/false positives and negatives, latency, etc.
 - » For mitigators: reduction in attack impact, ability to identify sources, etc.
 - Metrics for measuring defense cost
 - » Resources consumed – CPU/memory, bandwidth,
 - » Inadvertent harm caused to benign traffic - loss, delays
- **Metrics require instrumentation**
 - Should be independent of the mechanism under test
 - Requires positioning, calibration, interpretation of collected data
- **Key EMIST challenge: Provide guidelines for choosing metrics and instrumentation appropriate to individual experiments**



Introduction to our DDoS metric framework

- **Distinguish between extrinsic and intrinsic metrics**
 - **Extrinsic:** Measures that can be computed and observed by external parties
 - **Intrinsic:** Measures that can only be computed by the object being measured and by only analyzing the internal algorithms and data structures
- **Analyze metrics at different levels of abstraction**
 - packet, flow, aggregate, service/application layers
- **Metrics from different vantage points**
 - Client-side, server-side (end point), link-level, end-to-end
- **Focus on metrics for**
 - Traffic and service characterization
 - Measuring DDoS impact
 - Measuring DDoS effectiveness



Examples of Metrics for Characterizing Traffic (base traffic metrics)

	Client-observed	Intermediary-observed link level	Server-observed
Application-level	<ul style="list-style-type: none"> - streaming video: mean opinion score (MOS) (E) - VoIP: Round-trip-delay (E) - VoIP: Percentage of Packets discarded by the Jitter Buffer (I) - VoIP: Mean-length-of-bursts (I) 	Number-of-flows-per-application (E)	
Aggregate-level		- per-aggregate-packet-rate (E)	- per-aggregate-arrival-rate (E)
Flow-level	<ul style="list-style-type: none"> - server-connection-completion-time (E) - server-connection-completion-rate (E) 	- per-flow-packet-rate (E)	<ul style="list-style-type: none"> - per-client-connection-request-rate (E) - per-client-goodput (E)
	Rate-of-failed-connections (E)		
Packet-level	<ul style="list-style-type: none"> - server-response-rate (E); - server-response-time (E); 	<ul style="list-style-type: none"> - goodput (E); - ratio of attack traffic to goodput 	<ul style="list-style-type: none"> - per-client-packet-rate (E); - packet-drop-rate (I); - per-packet-processing overhead (I)

6. Network Services Infrastructure

- **DDoS attacks may have indirect affects on infrastructure services**
 - Cascading increases in DNS queries
 - Routing instabilities, esp. in BGP
 - Increased ICMP and ARP packets stimulated by spoofed addresses
- **DDoS attacks may directly target DNS servers or routers, e.g., Oct 2002 ICMP attack on 13 root name servers**
- **DDoS experiments that include infrastructure must specify infrastructure configuration in detail**
 - DNS servers, zones, relationships,
 - BGP autonomous systems (AS) and relationships
 - Intra-AS routing protocols and placement of routers
 - DHCP servers
- **Programmatically generating topologies and configurations**
 - Routers in AS topologies
 - Routing configurations
 - DNS configurations



7. Risk

- **DDoS experiments present low-to-moderate risk to the Internet**
- **Primary potential risk is that packet floods from attack generators could leak or backscatter onto Internet**
 - DETER security architecture addresses this risk
 - Potential leakage would have to emanate from small number of egress points that would be relatively easy to identify and shut off externally
- **Experiments with worms having DDoS zombies as payloads pose significantly higher risk**
- **Malcode Containment: Demonstrated DETER testbed ready to conduct Worm-spreading-DDoS zombie experiments**



Plans

- **Complete initial experiments, focusing on the experimentation process**
 - Analyze data, assess its value
 - Refine test conditions and instrumentation
 - Improve experiment automation
 - Generate recommendations and lessons learned
- **Develop better models of topology, traffic, attacks, etc. (Collaboration across EMIST teams.)**
- **Enhance tools, esp. for generating/manipulating traffic**
- **Continue outreach, solicit experiment proposals from external research groups and product teams**
- **Calibration: Scientific Instruments Require Fine-tuning**



Questions

- **Come talk to us at the poster session!**