Isolated virtualised clusters:

Testbeds for high-risk security experimentation and training

José M. Fernandez (*)
École Polytechnique de Montréal

Information Systems Security Research Lab
(*Laboratoire SecSI*)

(*) Joint work with:
- Carlton Davis, Pier-Luc St-Onge – Lab SecSI, Montréal, Canada
- Joan Calvet, Wadie Guizani, Mathieu Kaczmarek, Jean-Yves Marion – LORIA, Nancy, France

Agenda

- The Problem and the Objective
- The History
- The Design Criteria
- Architecture Description
- The Accomplishments
- Lessons Learned
- Future Work

Definition(s)

CSET

=

Computer Security Experimentation Testbed

Summary of Contributions

0. A very non-original and ambiguous acronym…
1. An alternative approach for CSET
   - ➔ Isolated virtualised clusters
2. A proposed list of design criteria for CSET
3. Conducting some “first-of-a-kind” really cool experiments
   - ➔ In-lab Botnet re-creation (3000 bots)
   - ➔ In-lab training of security grad students
4. Some lessons learned about building/operating a CSET

Why a CSET ?

- Trying to bring some of the benefits of the scientific method to Computer Security R&D
- In particular
  1. Experimental Control
  2. Repeatability
  3. Realism
- In contrast with
  - Mathematical modelling and simulation
  - Field experimentation

Desiderata and challenges of a CSET

- From CSET Workshop CFP
  - Scale
  - Multi-party Nature
  - Risk
  - Realism
  - Rigor
  - Setup/Scenario Complexity

Risks of CS R&D and CSET

- Confidentiality
  - Privacy of data (e.g. network traces)
  - Details of “real” system configurations
  - Security product design features
  - High-impact vulnerability information
  - Dual-use tools and technology (e.g. malware)
- Integrity and Availability
  - Effect on outside systems
    - University computing facilities
    - Internet

The SecSI/LORIA Story

Lab SecSI

École Polytechnique, Montréal

- 2005
  - Initial design and grant proposal to Canadian Foundation for Innovation (CFI)
- 2006
  - CFI Grant approved: 1.2 M$
- 2007-2008
  - Construction and eqpt acquisition
- 2009-
  - Tool comparative analysis & configuration
  - Initial experiments
  - First student projects
- 2010
  - First large scale experiments
  - Graduate course taught on testbed

Laboratoire Haute Sécurité

INPL/LORIA, Nancy, France

- 2007
  - LORIA and regional government support for LHS
- 2008
  - Collaboration starts with Lab SecSI
- 2009
  - Eqpt acquisition & config
- 2010
  - Official launch 1 July

Risk Management Measures

1. Self-imposed Laboratory Security Policy
   - Strong physical security
     - “Onion” model
     - Separate access control & video surveillance
   - Strong logical security
     - “Air gap” whenever possible
   - Personnel security

Risk Management Measures

2. University-imposed Review Committee
   - Aims at reducing computer security research-related risks
   - Tasks
     - Evaluates risk
     - Examines benefits of research against risks.
     - Examines and vets counter-measures and project
   - Includes external members and experts

➔ Not imposed by research granting-agencies

CSET design criteria

In order to achieve overarching goals of

- Realism
- Scale
- Flexibility

We defined the following criteria ➔

1. Versatility
2. Synchronisation
3. Soundness
4. Transparency
5. Environment
6. Background
7. High-level Exp. Design
8. Deployability
9. Manageability
10. Portability
11. Sterilisability

Isolated Virtualised Clusters

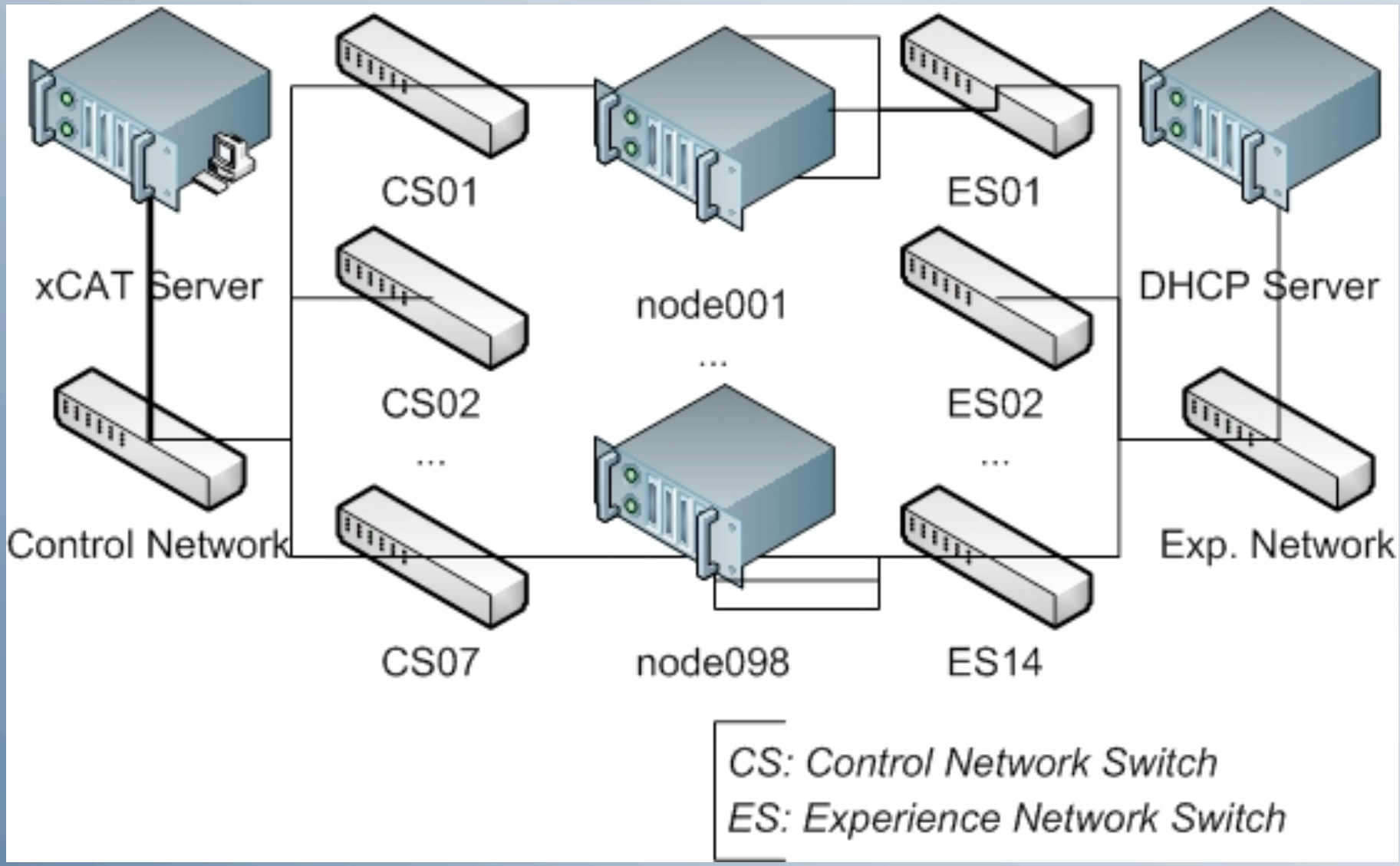
Isolated

- Research programme required high-risk experiments
- Lack of control on typical network-layer isolation measures
- Tried to follow model of Government of Canada security policy and IS security policy

Virtualisation

- Scale, scale, scale !!
  - Emulated machine typically does not require much CPU
  - Test conducted showed typical machine could support 50-100 VM
- "Built-in" manageability and portability
- Challenges/questions
  - VM/host isolation
  - Versatility
  - Cost

Network Architecture



Baby & Mumma Cluster

“Baby”

- 14 machines
- Used for
  - Student training
  - Experiment development
  - Low-risk experiments
  - Experiments requiring network connectivity
  - Very high-risk experiments (before and after sanitisation)
  - Increasing “Mumma”’s firepower

“Mumma”

- 98 machines
- Used for at-scale experiments
- Always isolated
- Can be partitioned (air gap) for conducting simultaneous experiments
- Supporting infrastructure
  - Adjacent console room
  - 12 Tb file server

Management tools

- Considered two options: DETER and xCAT
- xCAT
  - “eXtreme Cluster Administration Tool”
  - Open-source, initially developed/supported by IBM
  - VMWare ESX support initially custom-developed, now mainstream
  - Allows deployment and management of VM as if they were real nodes
  - Allows high-level design with VM as design element (higher granularity)

Design methodology

- Higher level design
  1. On paper high-level environment design
  2. Generate VM images for each machine type
  3. Write Perl scripts to generate xCat tables (as per design)
- Deployment
  - Run xCat scripts ➔ deploys and configures all VMs in a few hours
- Network configuration
  - No ability to generate switch configuration (yet)
  - Manual network configuration (patch panel/switch)
- Measurement & Monitoring
  - Custom monitoring/measurement application run on VM
  - Network traffic sniffing
  - VM management tools

Achievements - SecSI

1. DDoS experiment
   - Study of DoS resilience of various SMTP servers
   - 50 machines, run “on-the-metal”
2. Waledac Botnet Experiment
   - Recreated complete Waledac C&C infrastructure
   - Sybil attack experiment on 3000-bot Waledac
3. Graduate Security Course
   - Mandatory worm-experiment lab assignment
   - 2x from-scratch class projects
     (IDS & “concept” botnet)

Lessons Learned

- There is a lot to learn from high-scale, high-risk experiments in isolated testbeds …. (Wow!)
  - It cannot be learnt by other methods (e.g. in-the-wild experiments)
  - It is less risky…
- Disadvantages
  - Access by researchers complicated
  - Experiment design and testing more arduous

  ➔"baby" cluster not a luxury…

Lessons Learned

- Virtualisation
  - Larger scale, more flexibility
  - Deployment and monitoring not supported by all toolkits (e.g. DETER)
  - Some experiments still need to be run on-the-metal (synchronisation)

Achieving CSET design criteria

1. Versatility
2. Synchronisation ???
3. Soundness
4. Transparency ???
5. Environment
6. Background
7. High-level Exp. Design
8. Deployability
9. Manageability
10. Portability
11. Sterilisability ???

Future Work

1. Investigate/manage risk of VM containment failure
2. High-level design
   - More intuitive tools (vs. Perl scripts)
   - Granularity to the process/programme
3. Environment
   - Include network topology in high-level design
   - Automated network configuration deployment ("a la" DETER)
4. Background
   - A whole other topic in itself….
5. Make a cool DVD....