

Your risk is not what it used to be



Ariel Waissbein
-Core Security Technologies-



- Lic. in Maths ---> Ph. D
- Early research @ uba.edu.ar
 - theoretical computer science
 - computer algebra
 - elimination theory
 - algebraic geometry
- Enters computer security @ Core
 - crypto
 - vulnerabilities
 - software protection
 - web application & endpoint security
 - pentesting

In seeking wisdom, the first step is silence;
the second, listening; the third,
remembering; the fourth, practicing; the
fifth teaching others.

Solomon Ibn Gabirol, Poet and Philosopher (c. 1021-1058)

web administrator

Pentester

Security Officer

We? Who's we?

Network administrator

Chief Operations Officer

Chief Technology Officer

Chief Security Officer

Web application's developer

Database administrator

- Intro / motivations
- Contemplations
- Learn about the exploit
- Can hackers do what?
- Browsing MyLog
- Preventing the same mistake
- Wrap-up



- 20 years ago
 - vulnerabilities were reported to a closed circle and
 - only later patches (or countermeasures) were slowly pushed
 - Zardoz security list (1989-1991)
 - Core security list (1990-1991)
- Today,
 - we use more software
 - patches are issued daily
- But, what have we learned?

No patches nor security updates means that you are uninformed



Did that exploit
hurt you?



Did that exploit hurt
you?

When information for
patched vulnerabilities isn't
used to determine threats,
threats are ignored!



The habit of reading logs.

New information helps to surf the log trails associated to real threats



Improving also means not committing the same mistakes. So you must make sure you don't



- Intro / motivations
- **Contemplations**
- Learn about the exploit
- Can hackers do what?
- Browsing MyLog
- Preventing the same mistake
- Wrap-up



- Daily we get info about
 - patches & advisories
 - security incidents & malware

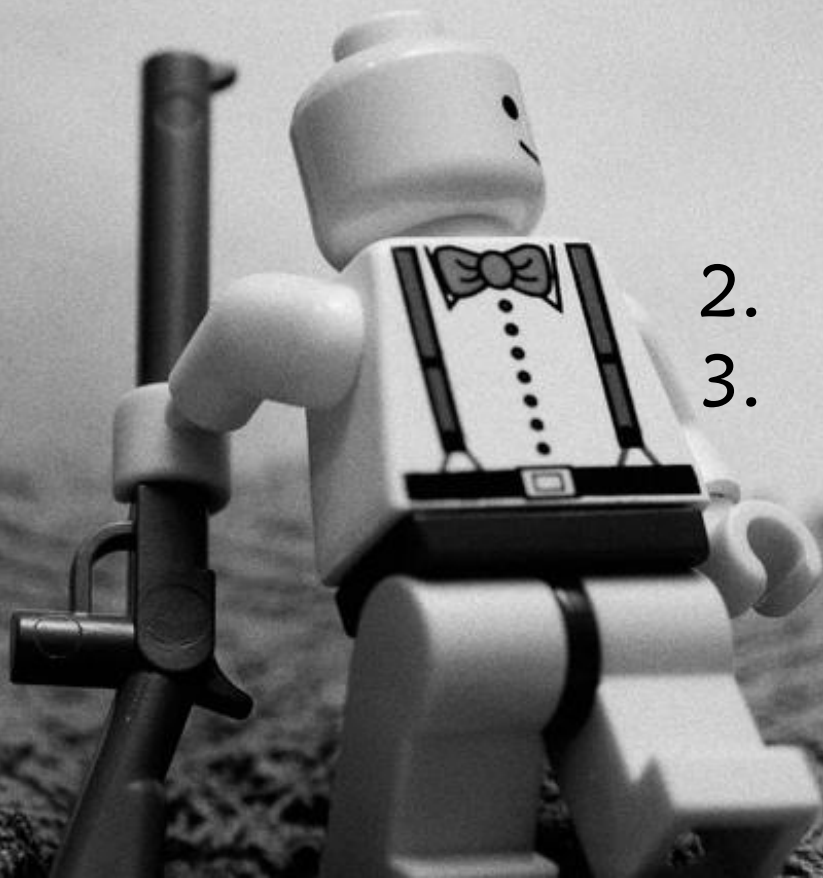


- We can use this info to our advantage
 - Yet: how can we make accurate calls and minimize risk and gambling?

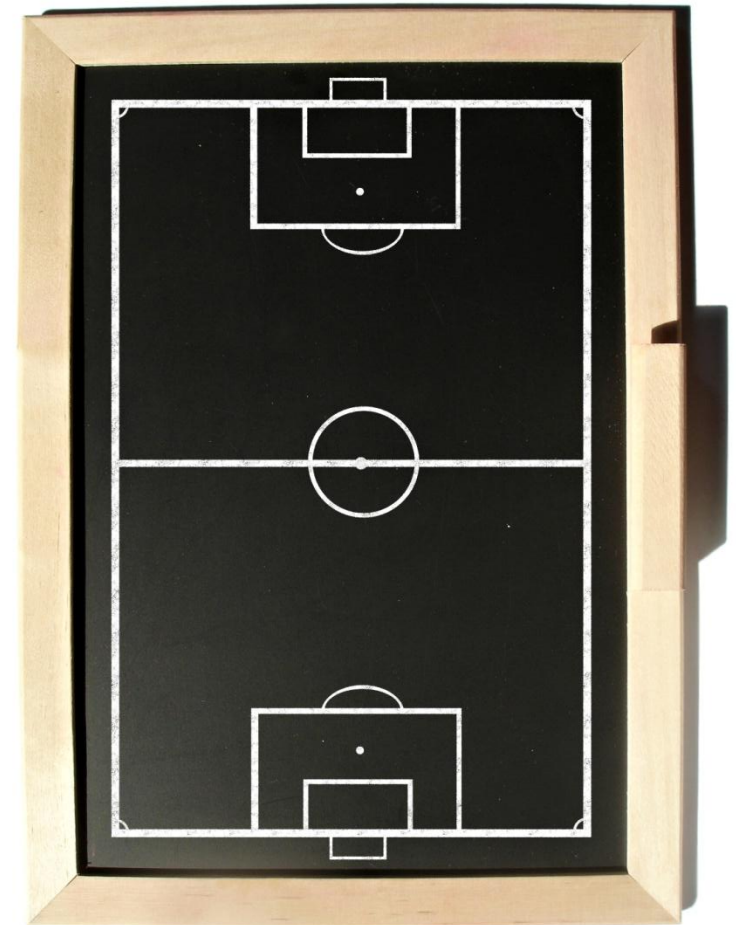
Play with toys

We can go through the following methodology:

1. Scan the network to document computers (their OS, services, apps, ...), networking and vulnerabilities
2. Copy this in a blackboard
3. Use the info from the exploits to draw attack paths (and envision threats)



- Little did we know ...
(about 1,2 and 3)



- Little did we know...
- In terms of security you must
 - know which exploits work against what
 - check potential attack paths
- How can you do that?
 - Many have tried but using inaccurate models



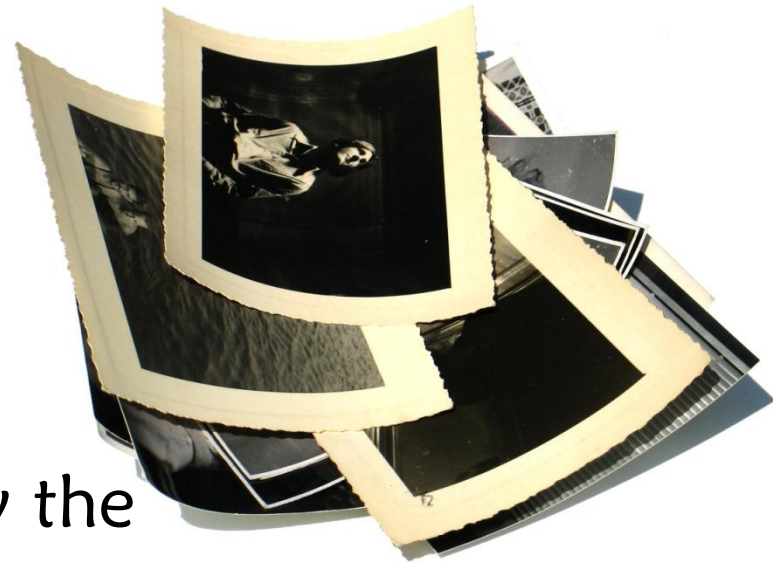


Let me suggest a way to do this...

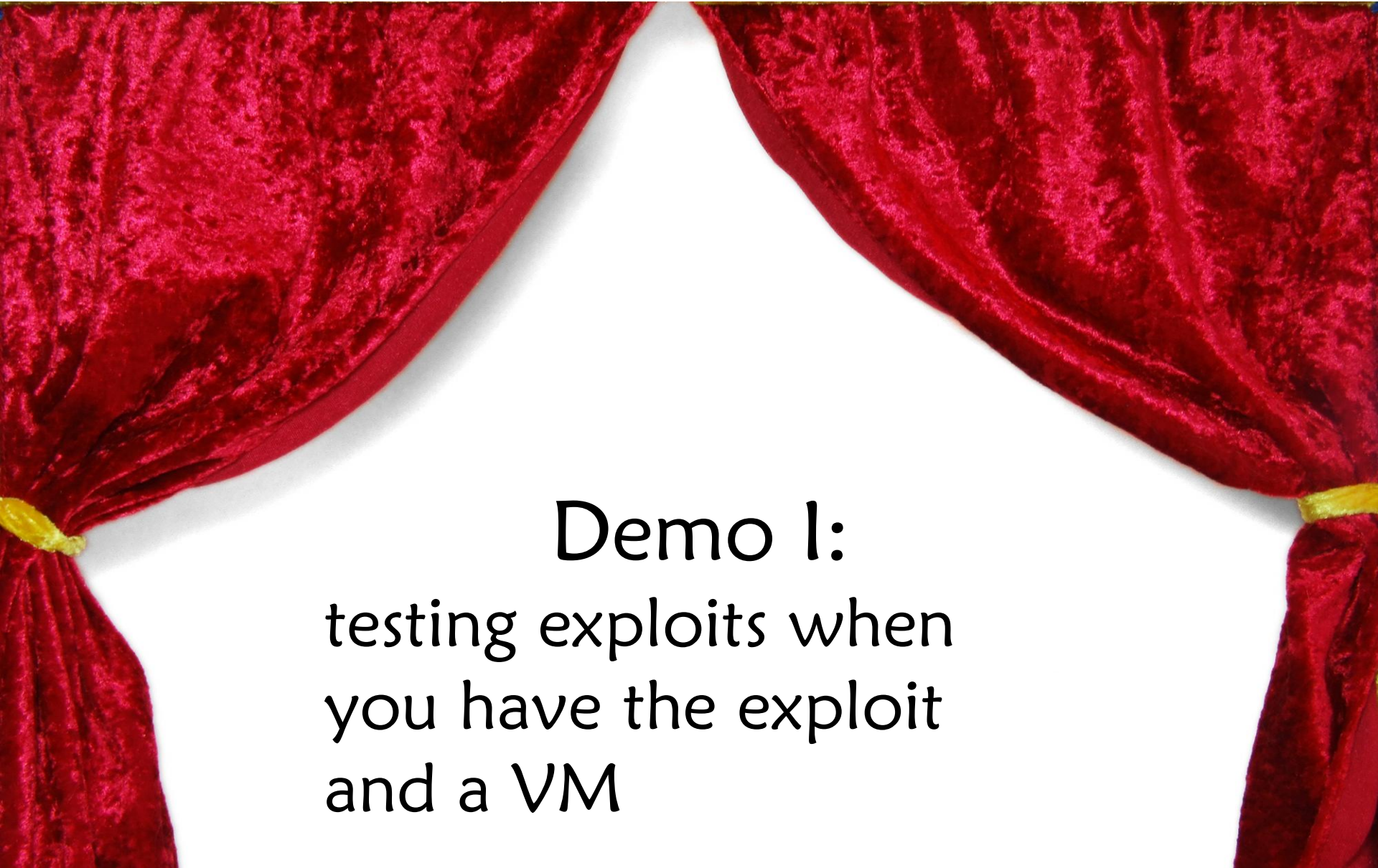
- Intro / motivations
- Contemplations
- **Learn about the exploit**
- Can hackers do what?
- Browsing MyLog
- Preventing the same mistake
- Wrap-up



- Targets
 - Keep installers and updates for all your software, or
 - Maintain a virtual machines DB with snapshots from
- Exploits
 - Get the exploits (you know the sources!)
 - Check them against these VMs



Caveat: if you do not have the resources, there is still place for “accurate speculation”

A pair of red velvet curtains pulled back to reveal a white background. The curtains are tied back with yellow ribbons.

Demo 1:

testing exploits when
you have the exploit
and a VM

- As attackers can develop exploits and gather statistics so must we
 - If a vulnerability could be exploited even if a protection is enabled (e.g., ASLR = address space layout randomization), then it will
- Testing involves using the exploit against different targets
 - E.g., OS, service pack, protection settings, additional appls & services, languages

- You can get the picture if you get an exploit for the same vulnerability.
- At least you'll know which systems are exploitable.



| Target version | Exploit | Result |
|----------------|----------------------------------|-----------------|
| 7.4.03.30 | SAP DB Web Tools Buffer Overflow | not exploitable |
| 7.4 | “ | exploitable |

Web Tools is a SAP DB module that allows you to manage the DB from a web browser

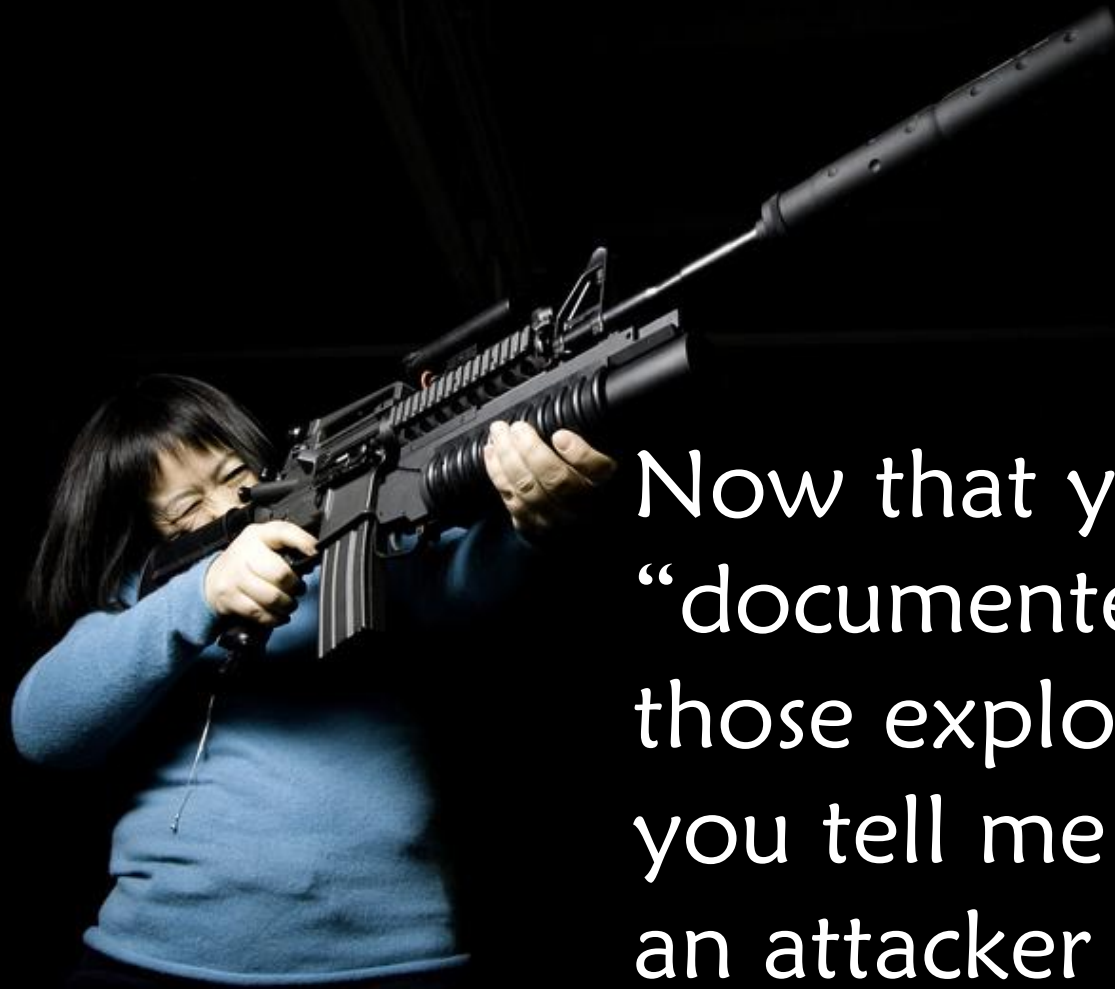
- Which versions are vulnerable?
 - What does the vendor say?
 - Is there an advisory?
 - Assume the worse



- Intro / motivations
- Contemplations
- Testing exploits
- **Can hackers do what?**
 - Insight simulation tool
- Browsing MyLog
- Preventing the same mistake
- Wrap-up



What are weapons for?



Now that you “documented” all those exploits. Can you tell me what can an attacker do?

Insight is a tool that interfaces with a pen-testing framework and simulates (attacks against) computer network scenarios

Futoransky, Miranda, Orlicki and Sarraute devised and implemented Core Insight (see [MOS 07]).

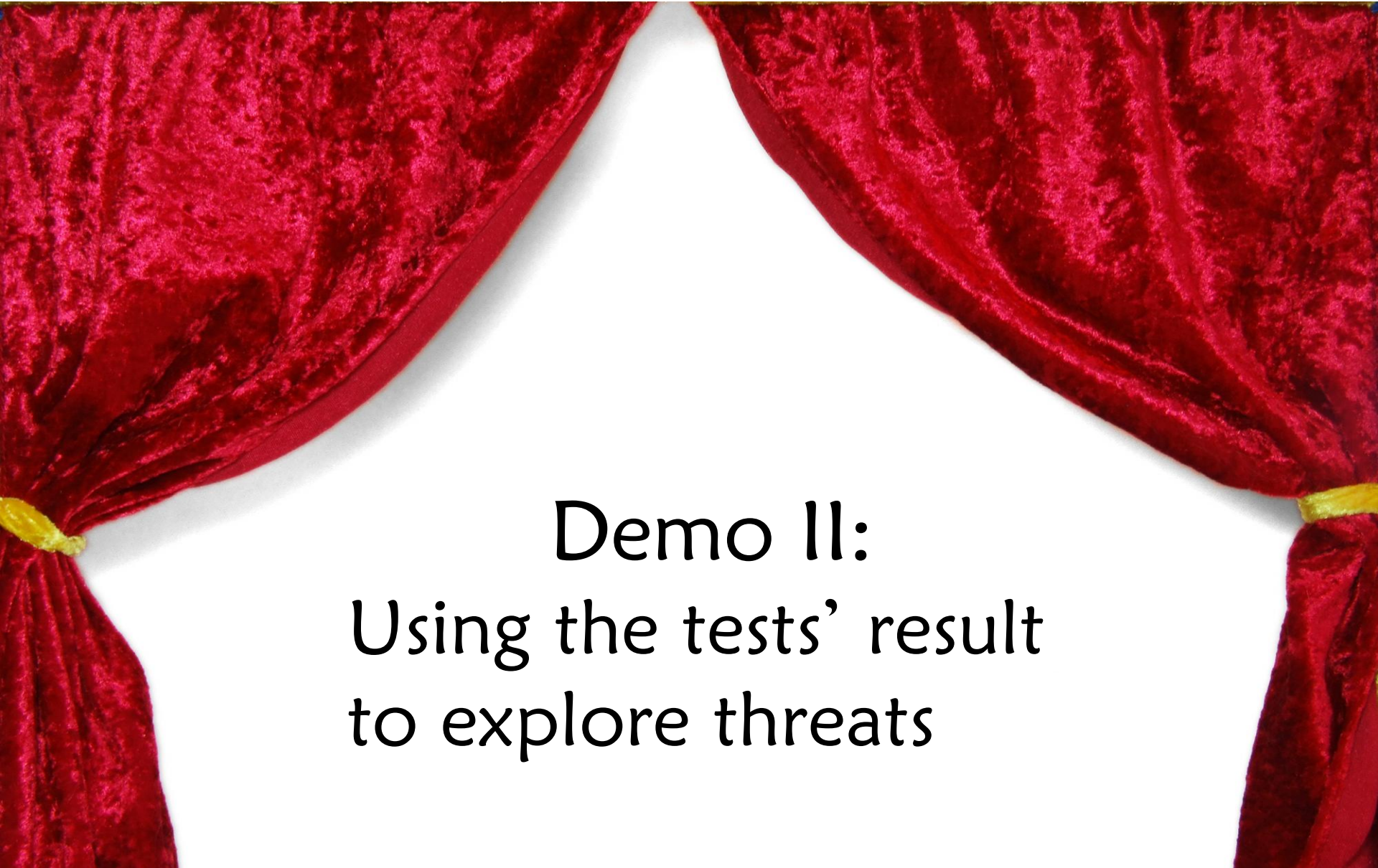


1. We take snapshots of the network's configuration

- Computers – OS – patch level
- Topology
- Vulnerabilities

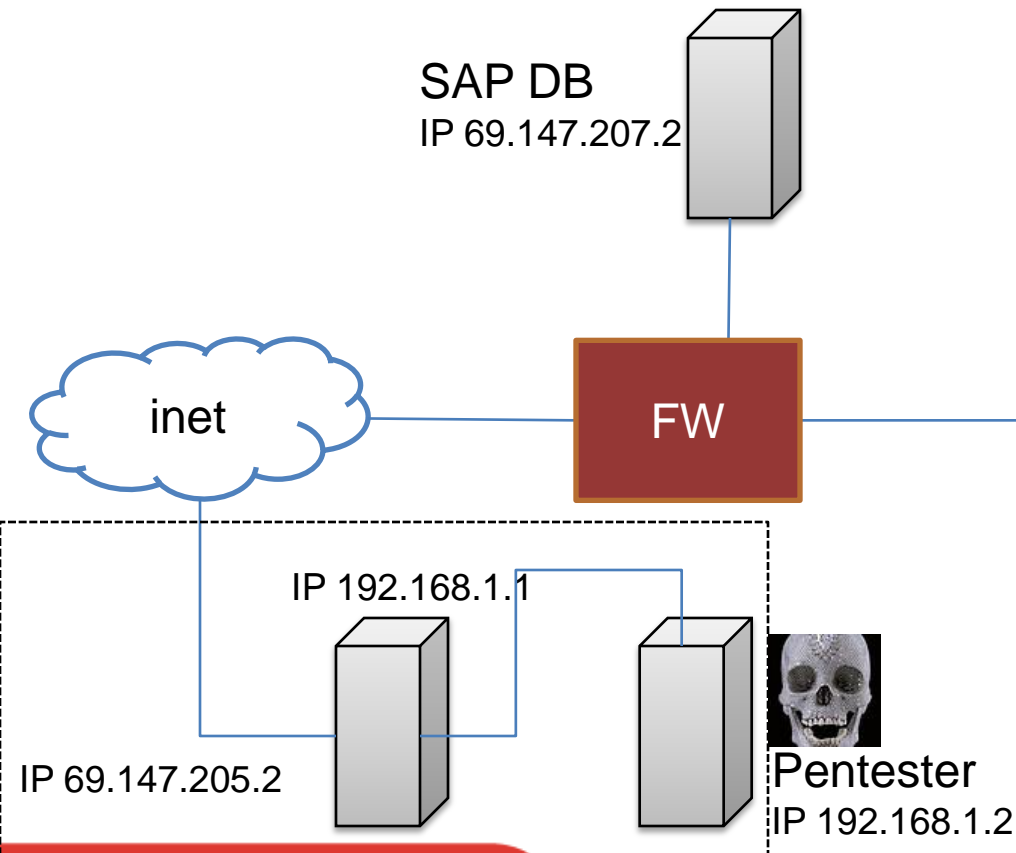
2. We “copy” these snapshots to our simulator

3. Now you can execute simulated attacks against each of these snapshots
(Even if the network changes)

A pair of red velvet curtains pulled back to reveal a white background. The curtains are tied back with yellow ribbons.

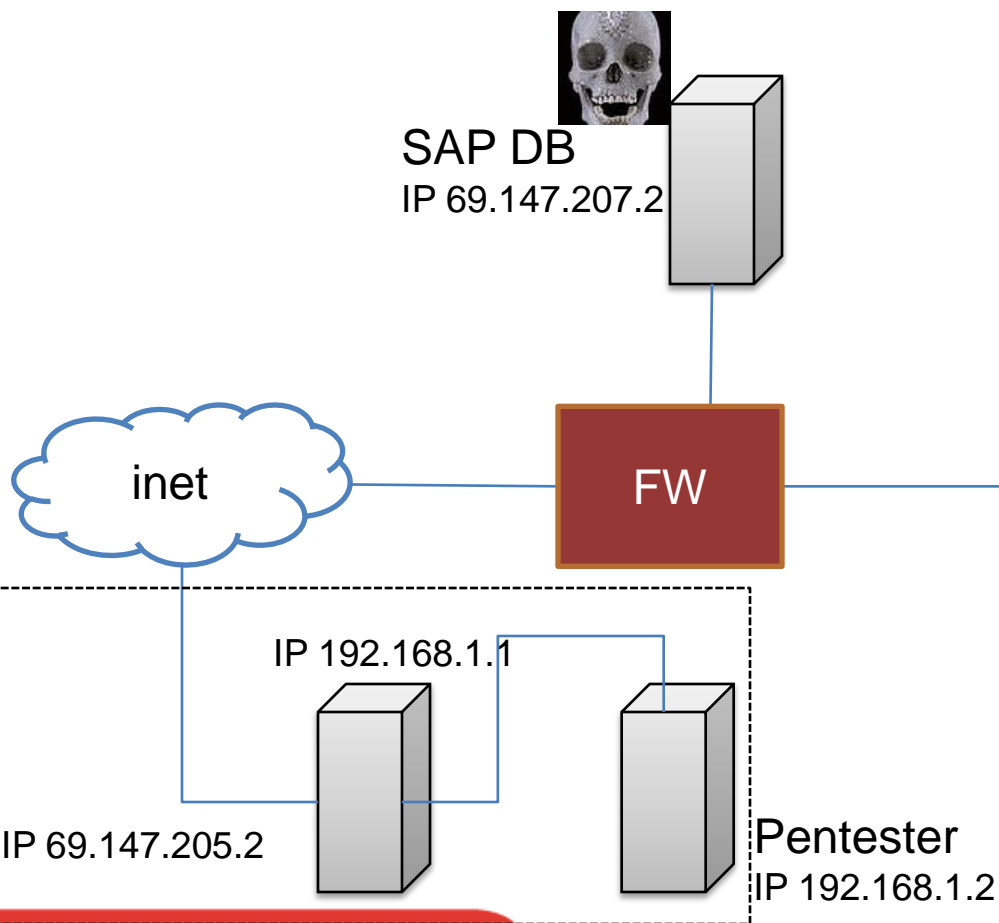
Demo II: Using the tests' result to explore threats

- In our imaginary network
 - we have a new exploit against SAP DB and
 - a little birdy said there'd be a SAP DB sitting in the DMZ



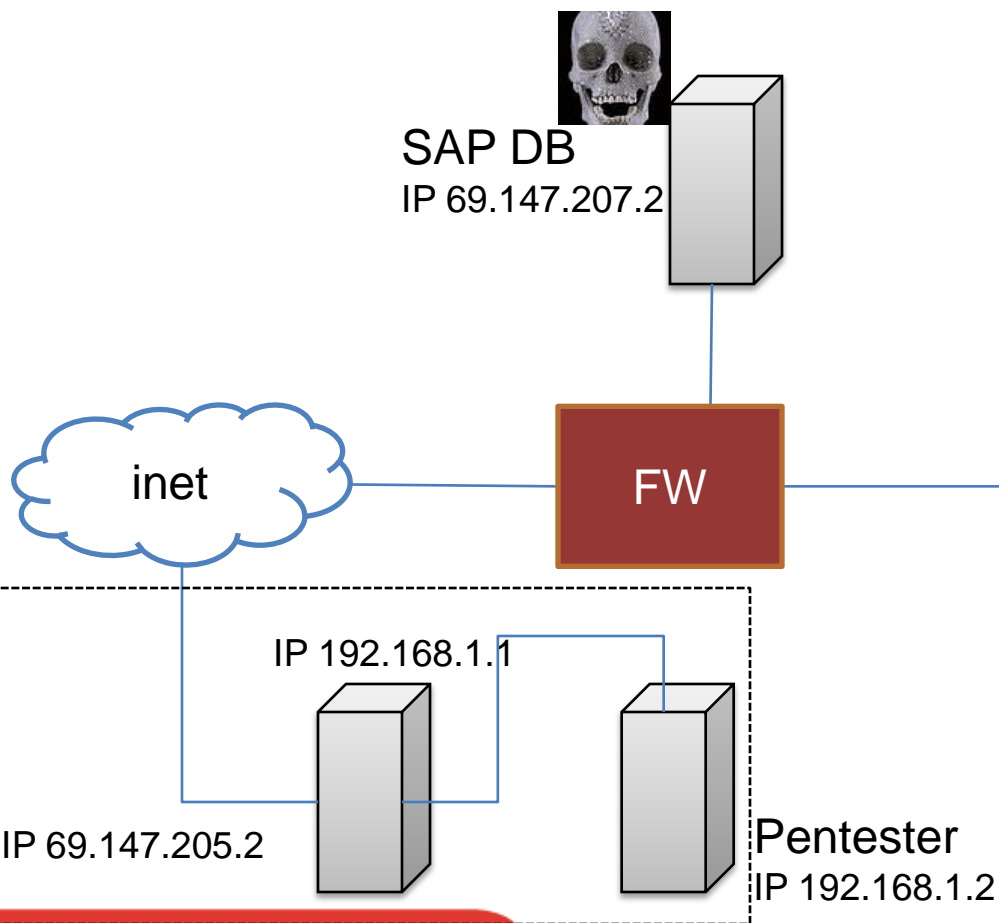
Whatever they
have in the
internal network

- We exploited the SAP DB server and set camp there
 - we do local IG, browse files and see a reference to IP 192.168.17.4



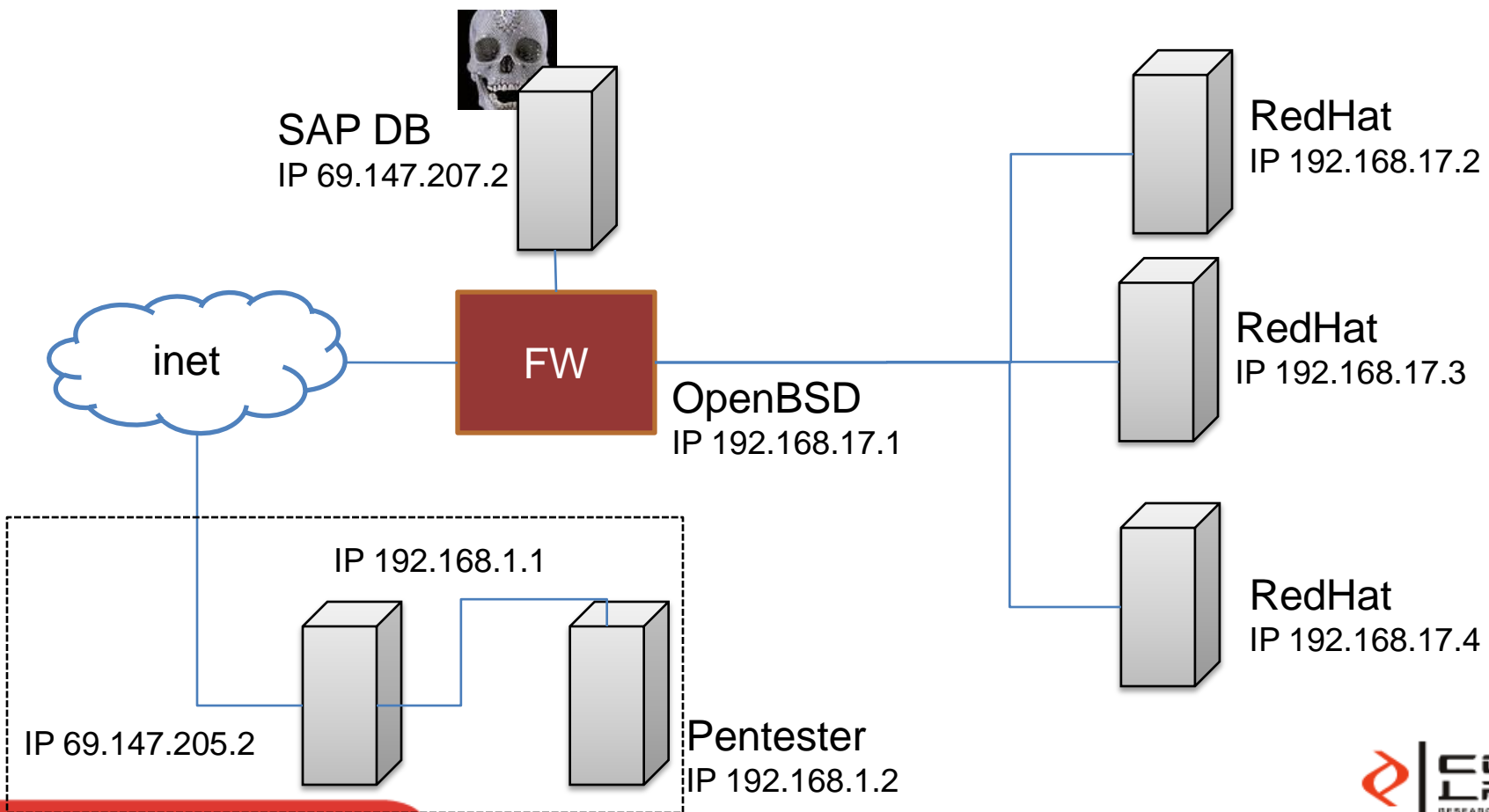
Whatever they have in the internal network

- We exploited the SAP DB server and set camp there
 - we do IG over the 192.168.17.* IP space and ...

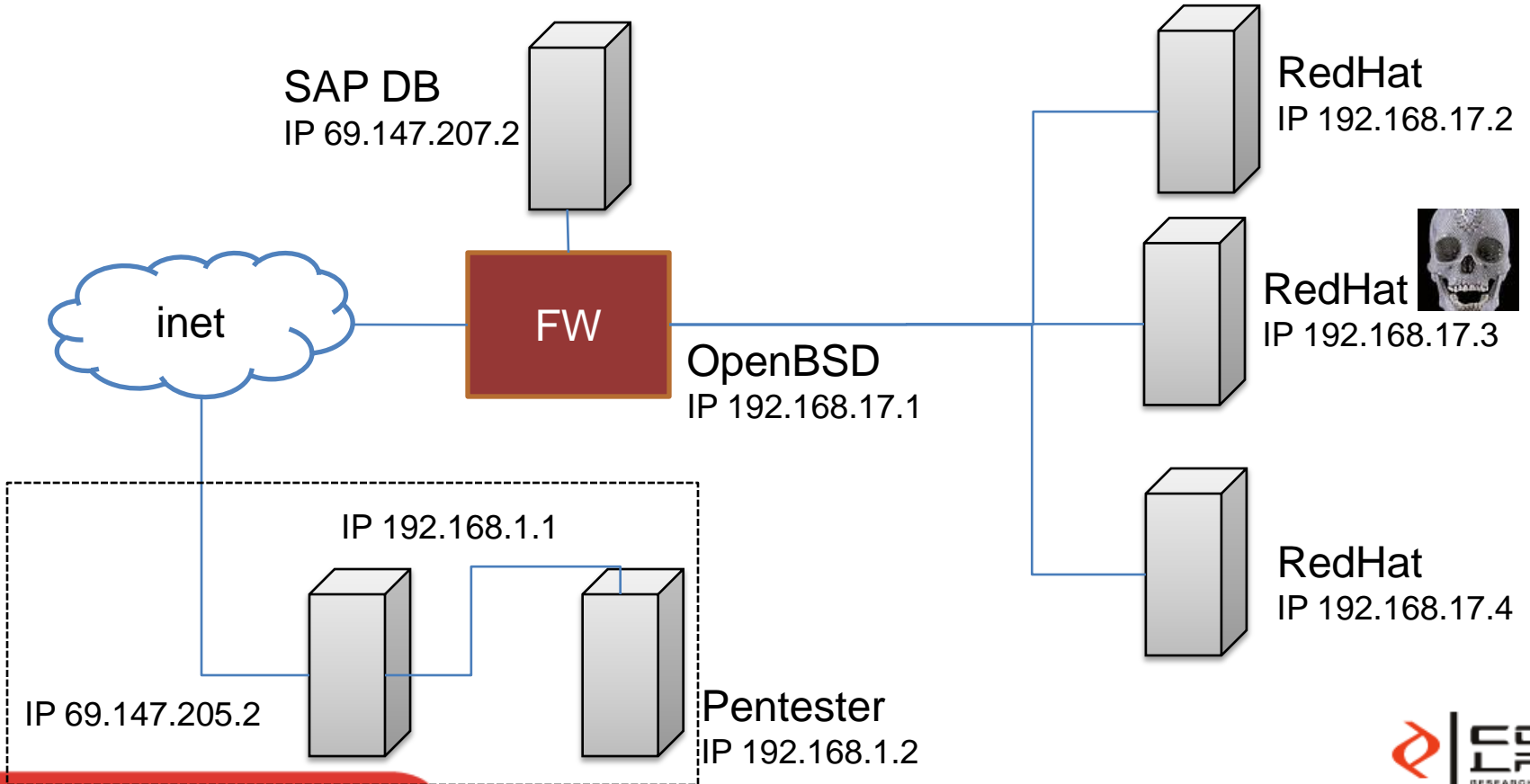


Whatever they have in the internal network

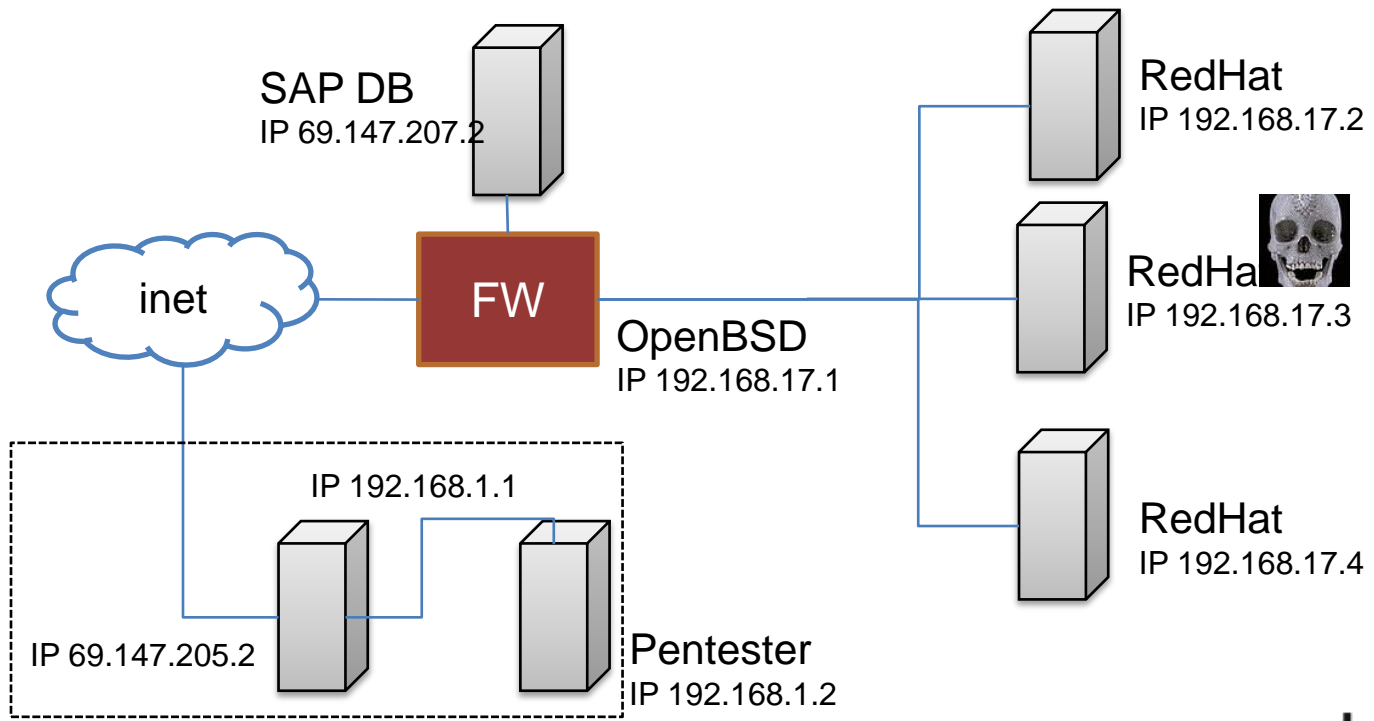
- We discover three RedHat desktops with different services running



- The RedHat in 17.3 runs a vulnerable OpenSSL
 - we hack into it, browse files and discover an encrypted credit cards database... and the password



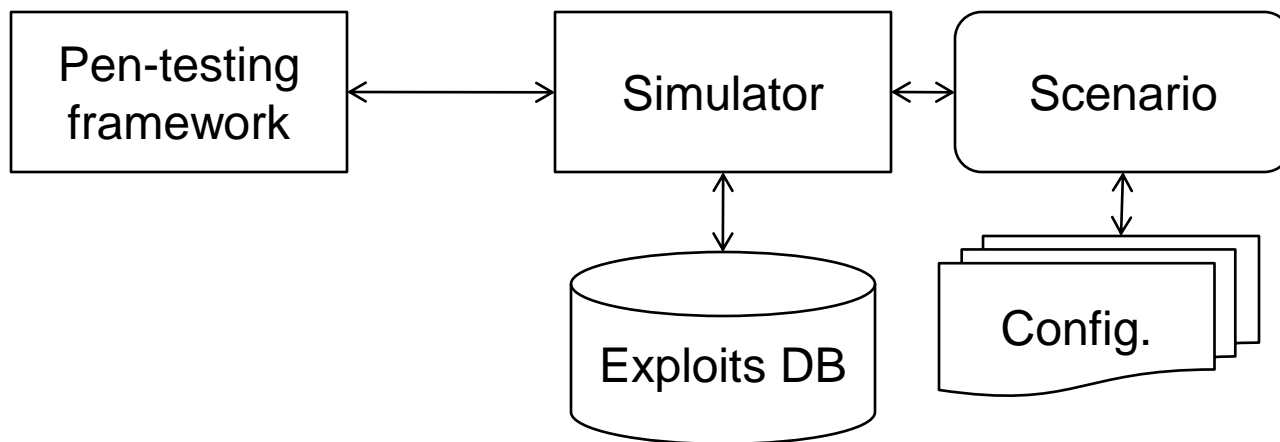
- The pen-test shows that *the vulnerable SAP DB* and the *firewall's configuration* allowed the attacker inside.



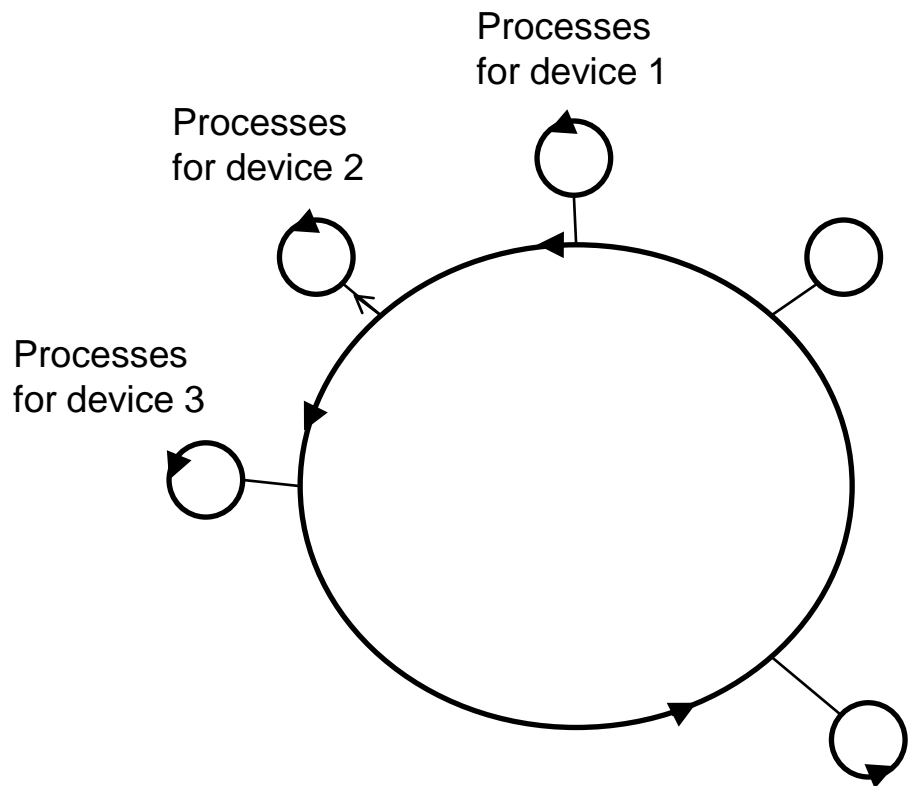
- Intro / motivations
- Contemplations
- Testing exploits
- Can hackers do what?
 - Insight simulation tool
- Browsing MyLog
- Preventing the same mistake
- Wrap-up



A device is simulated by its OS, file system, config. and vulnerabilities

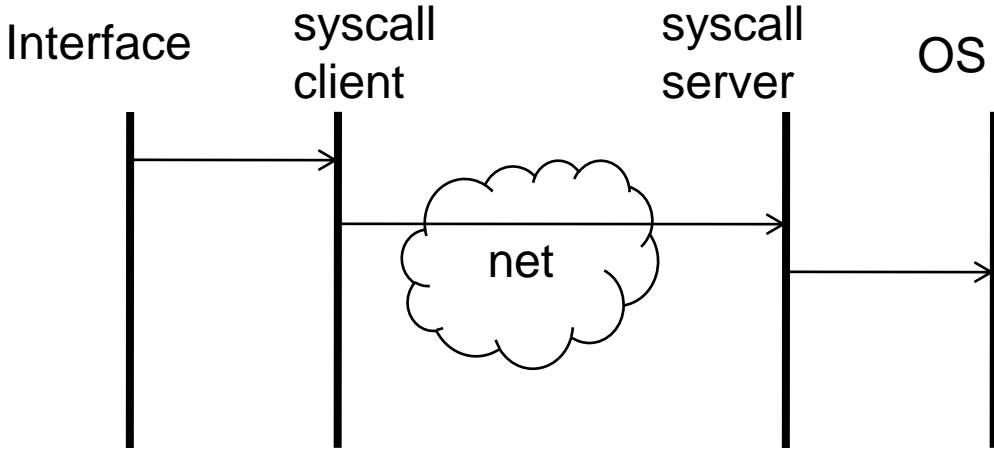


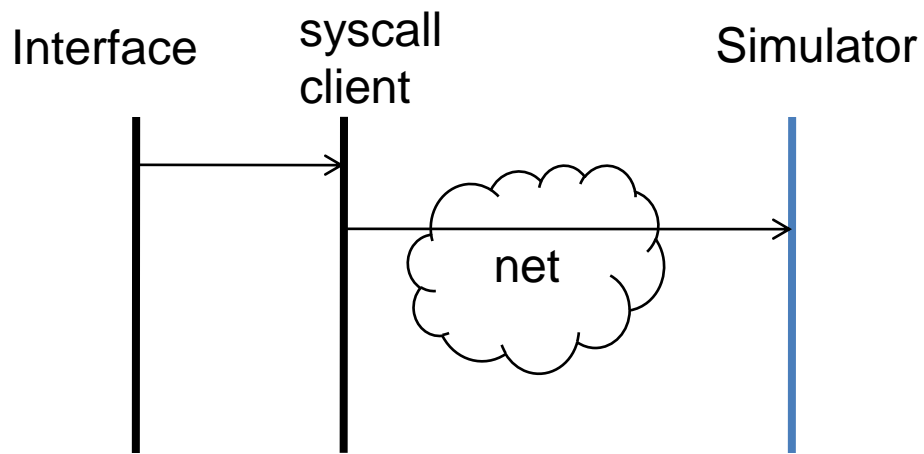
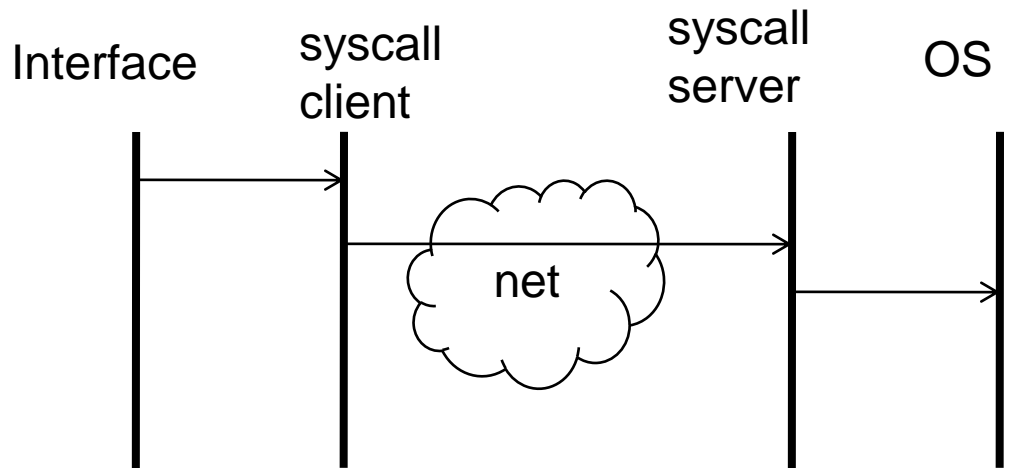
1. On startup the simulator starts one process per device in the config file.
2. Each device is started with certain services, processes and networking.
3. Simulated computers run syscalls.



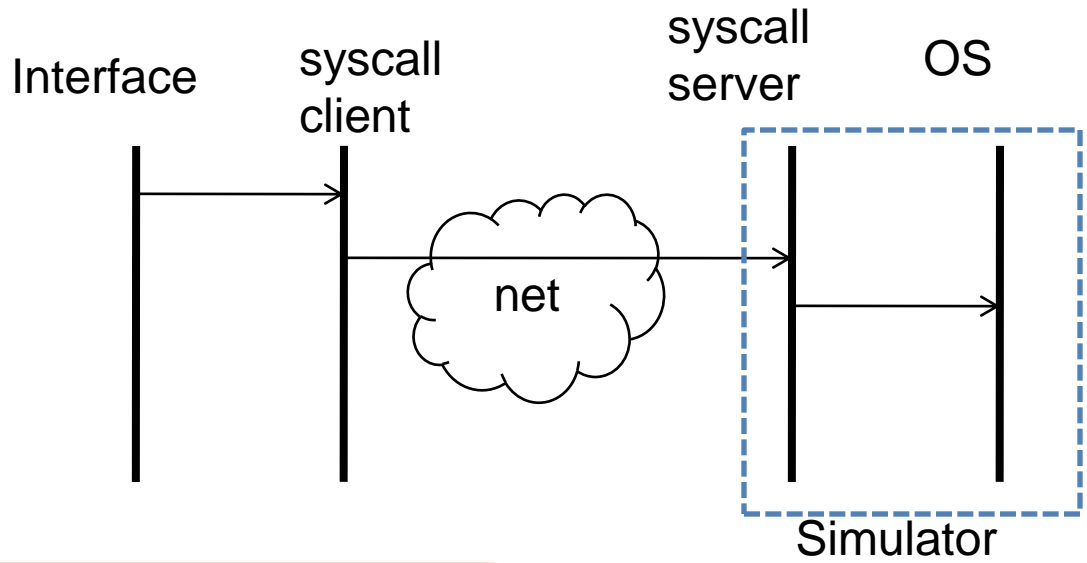
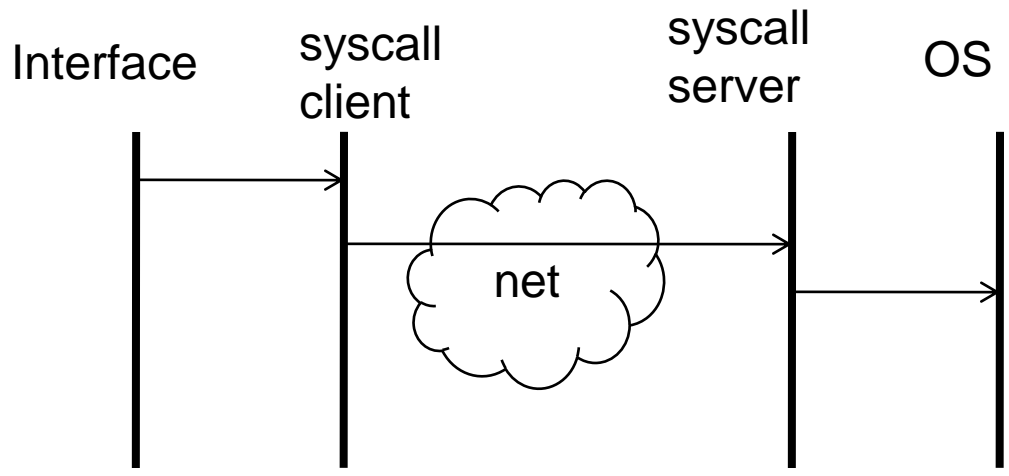
- One at a time, the simulator handles processes as demanded by config and then the pentesting framework
- Processes and actions are
 - simulated
 - or emulated

- We imitate the syscall proxying [AC01]





- We imitate the syscall proxying [AC01]
- The pen-testing fmwk issues the syscall and the simulator answers

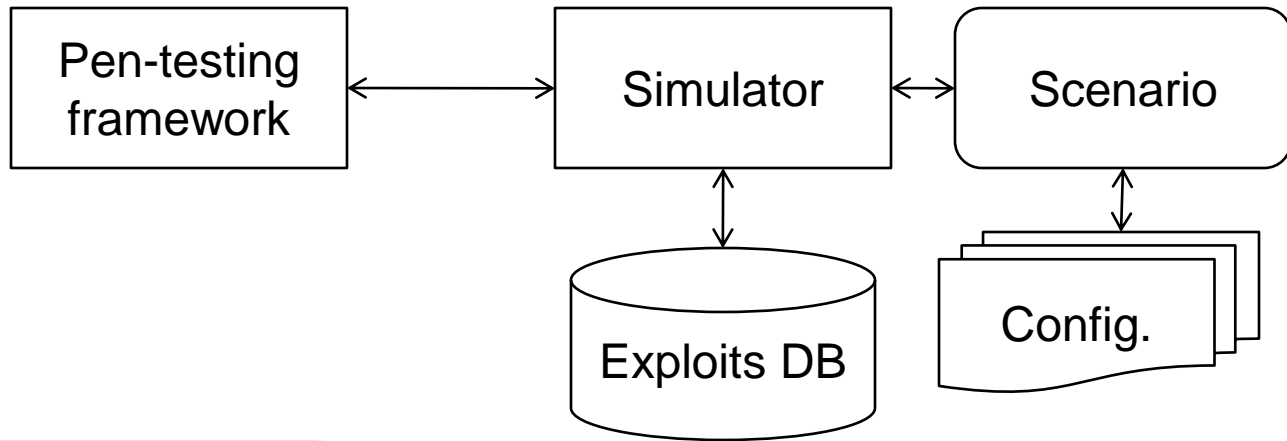


- We imitate the syscall proxying [AC01]
- The pen-testing fmwk issues the syscall and the simulator answers

Exploits are not simulated, their effect is modeled by a black-box procedure.



1. Exploit sent from PT fwkw.
2. Simulator queries DB
3. Simulator computes result and modifies scenario.
4. The scenario reflects an agent installed, or a crash or no answer.



- Scenarios are defined by
 - Computer / Network device
 - OS / service pack / ... (banners)
 - File system
 - Vulnerabilities

- Now, we do this through scripting
 - we do not have a “copy” functionality but a manual procedure

- Not simulated to full detail are:
 - network communications
 - exploits
 - some syscalls are not implemented
- This means that
 - some functions cannot be implemented and
 - all exploits must be *modeled*



- Intro / motivations
- Contemplations
- Testing exploits
- Can hackers do what?
 - Insight simulation tool
- **Browsing MyLog**
- Preventing the same mistake
- Wrap-up



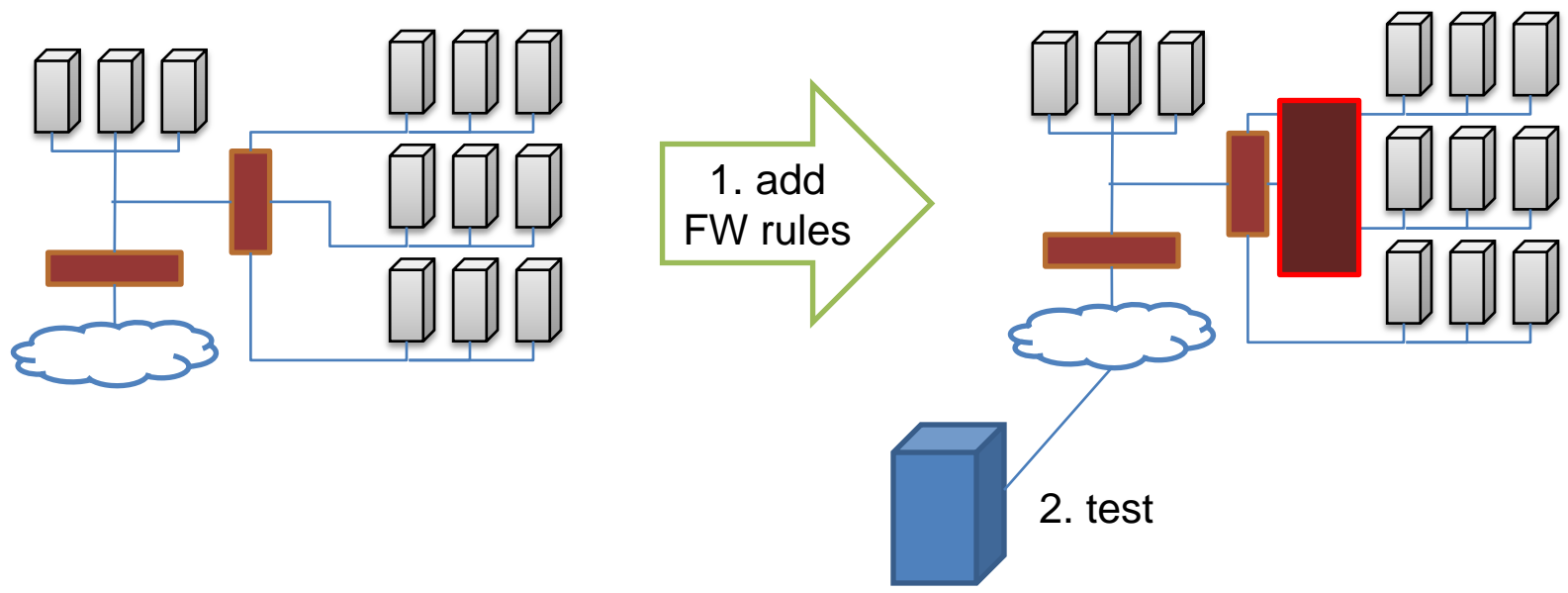
- Read the logs
 - you know the services exploited, the ports to look for, what events might be correlated
 - you know when they started and stopped being vulnerable

- There's great tools out there
 - and there's always grep

- Intro / motivations
- Contemplations
- Testing exploits
- Can hackers do what?
 - Insight simulation tool
- Browsing MyLog
- Preventing the same mistake
- Wrap-up



- You can simply modify the simulated scenario and test other settings



- Intro / motivations
- Contemplations
- Testing exploits
- Can hackers do what?
 - Insight simulation tool
- Browsing MyLog
- Preventing the same mistake
- **Wrap-up**



How do you know what's best for
your network's future?

You might not want to get married
with patches!

The network must evolve in a controlled manner

Could we design a process to do this?



Thanks!



A penny for your thoughts...?

- [MOS 07] Miranda, Orlicki, Sarraute “Simulation of Computer Network Attacks.” *AST 2007 - 36 JAIIO, Buenos Aires, Argentina, 2007*
- [AC01] Arce, Caceres “Automating Penetration Tests - a new challenge for the IS industry?” *Black Hat Briefings, Las Vegas, July 11-12, 2001*
- [Wai08] Weissbein “A Penetration Testing Learning Kit.” *Troopers 08. April 23-24, 2008. Munich, Germany*
- [LG05] R. Lippman, K. Ingols, “An Annotated Review of Past Papers on Attack Graphs” *Technical Report ESC-TR-2005-054, MIT Lincoln Laboratory*

- ❖ Agenda <http://www.morguefile.com/archive/?display=203210>
- ❖ Cow's skull by crydin <http://www.flickr.com/photos/crydin/367566323>
- ❖ Nurse by kafkan <http://www.flickr.com/photos/kafkan/189816757>
- ❖ Wall street bull by dougemoine <http://www.flickr.com/photos/kindee/289253349/>
- ❖ Death of a Loyalist Soldier, by Balakov <http://www.flickr.com/photos/balakov/1805926540/>
- ❖ Football blackboard <http://www.sxc.hu/photo/648105>
- ❖ Three cows <http://www.sxc.hu/photo/412289>
- ❖ Veleta <http://www.sxc.hu/photo/527407>
- ❖ Cut The Crap 2 by woodsy <http://www.sxc.hu/browse.phtml?f=download&id=808214>
- ❖ Girl with big gun by ttstam <http://www.flickr.com/photos/ttstam/2211800182>
- ❖ Diamond skull by secretlyironic <http://www.flickr.com/photos/secretlyironic/524919354>