ÓBUDAI EGYETEM

Advanced Vulnerability Assessment Tool for Distributed Systems (AVAT)

Sándor Ács, Péter Kotcauer
[acs.sandor, peter.kotcauer]@sztaki.mta.hu
Miklos Kozlovszky
kozlovszky.miklos@nik.uni-obuda.hu

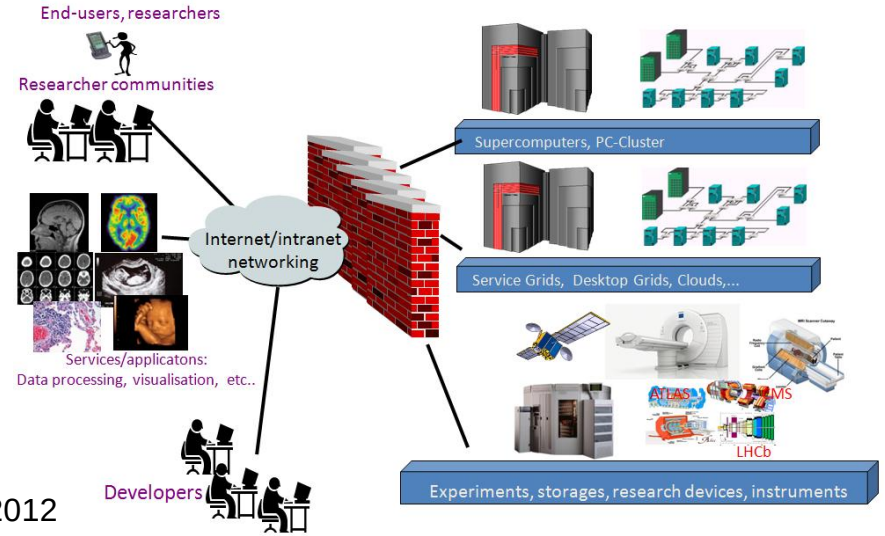
ÓBUDAI EGYETEM

Outline

- Motivations
- State-of-the-art
- OpenVAS
- Design and Implementation
- Vulnerability assessments results
- Conclusion and future work

Motivations

- Large scale resources
  - Supercomputing infrastructure, HPC centers, GRID/Cloud systems, etc.
  - Large impact if fails
- Homogeneous resources
  - Computational, networking, storage, middleware
  - Reusable break-in methods, large gain if successfull
- Lot of users/semi-open communities
  - Social networking issues
  - Weakest points



ÓBUDAI EGYETEM

Motivations (contd.)

- There are lot of potential security problems with distributed and shared systems:
  - Coming from the technology itself
  - Coming from site/softwarestack setup
  - Coming from end-user behavior
- General goal: Less vulnerable infrastructure (survive cyberattacks)
  - Protect user data
  - Eliminate malicious infrastructure (re)usage
- Our goal is to create a vulnerability assessment framework for distributed systems in order to decrease threats!

DCI monitoring around the globe (General DCI monitoring tools)

- Grid monitoring tools
  - SFTs (Site Functionality Tests)
  - Nagios
  - Ganglia
  - Netmon
  - PerfSonar
  - GridView
  - RTM
  - …
- Cloud Monitoring
  - Hyperic
  - Zenoss
  - RevealCloud
  - Rackspace Cloud Monitoring
  - …

Note: applications are often reused to monitor other DCIs

DCI monitoring around the globe (Security monitoring tools)

- Commercial vulnerability assessment tools
  - Qualys
  - Nessus
- Open vulnerability assessment tools
  - OpenVAS
  - Nexpose
  - Pakiti
    - EGI Pakiti
    - EGI Security Dashboard
  - GSSVA
    - SZTAKI solution some years ago used for SEE-GRID-SCI
    - Pakiti based
    - Only software stack investigation

ÓBUDAI EGYETEM

Issues with the existing solutions

- Mostly non-DCI compliant (or limited)
  - Non job-aware (job creation?)
  - Submission problems (middleware dependencies)
  - Result retrieval
  - Working in user space (how, and how not)
    - port opening for communication
- Service-like operation is limited
  - Hierarchical authentication schemes
    - available mostly in commercial solutions
  - Periodical monitoring
  - Alarming schemes
  - „Security sampling" solutions
    - Try not disturb the normal business

ÓBUDAI EGYETEM

OpenVAS

- The core of the **AVAT** (*Advanced Vulnerability Assessment Tool for Distributed Systems*) is based on Open Vulnerability Assessment System (OpenVAS) framework.
- OpenVAS:
  - It collects several services and tools to provide a vulnerability scanning and vulnerability management solution.
  - Free software (most components are licensed under the GPL).
  - External NVT (*Network Vulnerability Test*) repository.

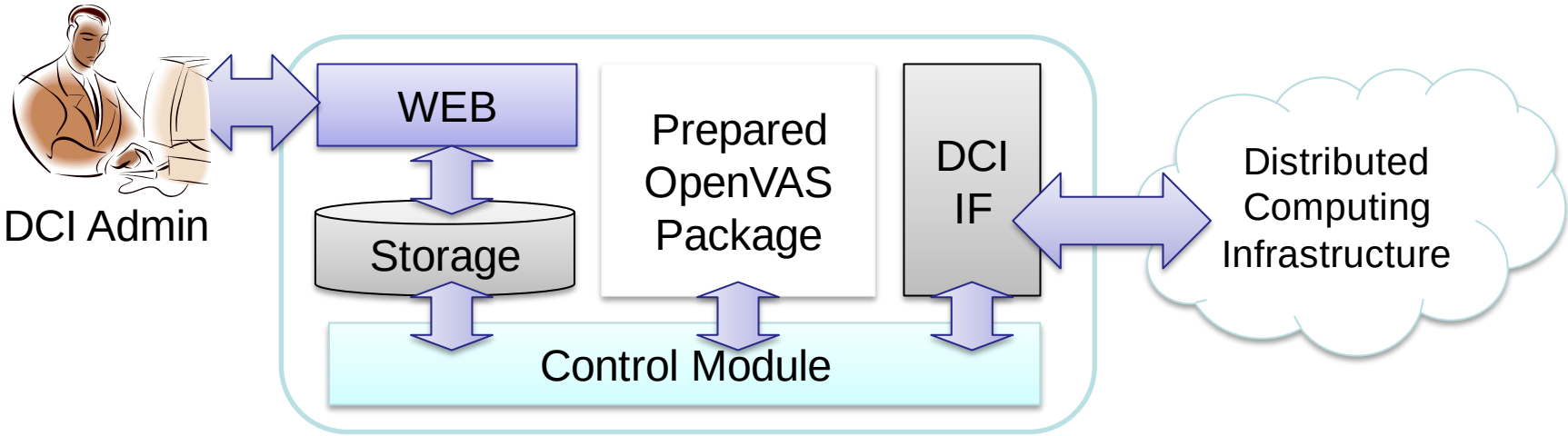
ÓBUDAI EGYETEM

OpenVAS from DCI perspective

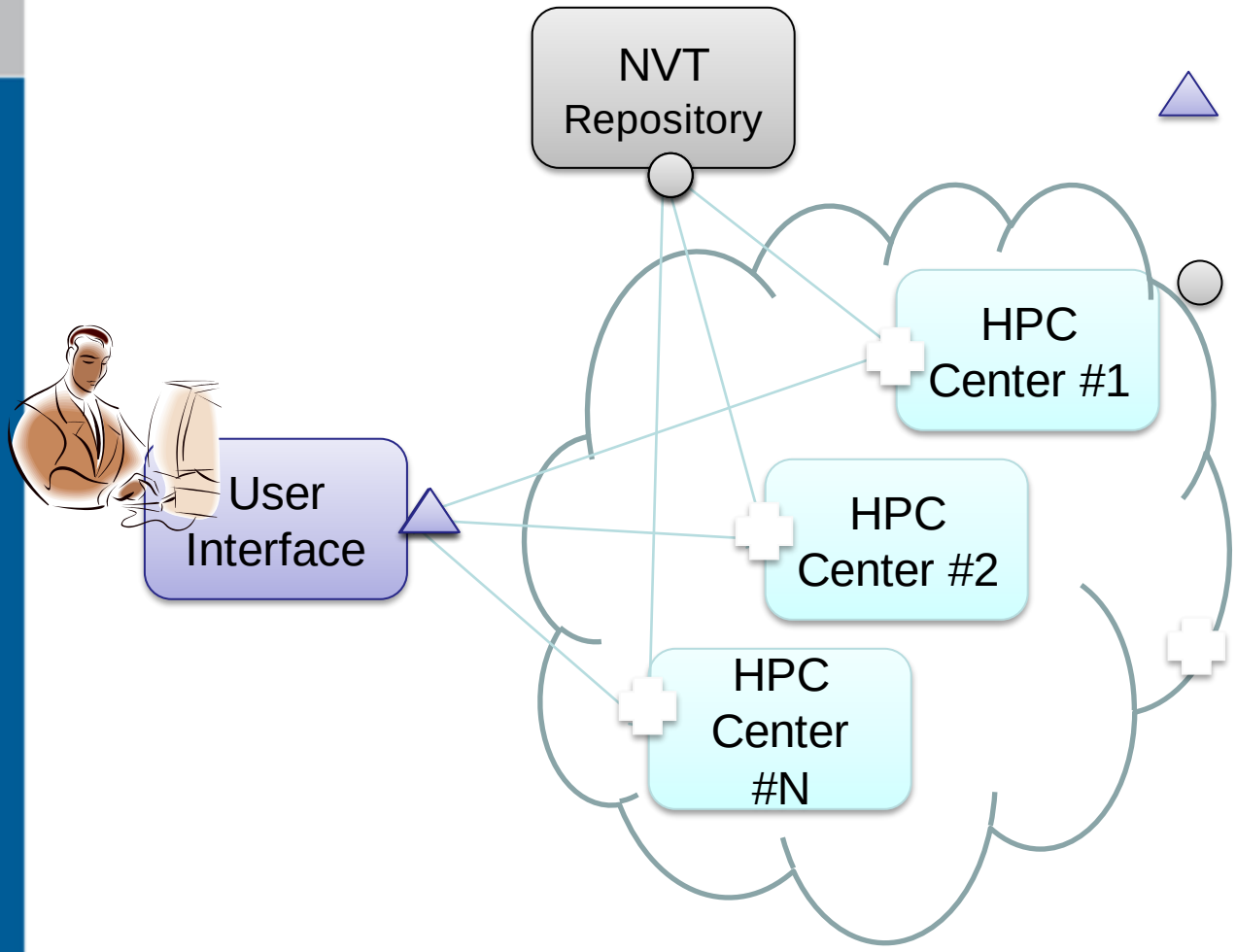
- OpenVAS Server: about 30 MB (packed
- OpenVAS Server: about 300 MB (unpacked)
- Working in user space after some tricks
- Communication via ssl/https
- Server and client on the same host
  - Localhost 127.0.0.0 range can be used
- Client output: html file
- One vulnerability assessment = 1 html result file
- Needs to collect, harmonize and visualize
- Server + client are running on worker nodes (but only computing elements are addressable)
- No direct submission is possible (use brokering services if available)

Design and Implementation (1)

- During the investigation, we have used:
  - ARC middleware for the centers of the HP-SEE infrastructure
  - gLite middleware for the SEEGRID VO.



Design and Implementation (2)



- Submit test jobs.
- Start the testjob and download NVT updates.
- Send back the result of the vulnerability scan to the UI.

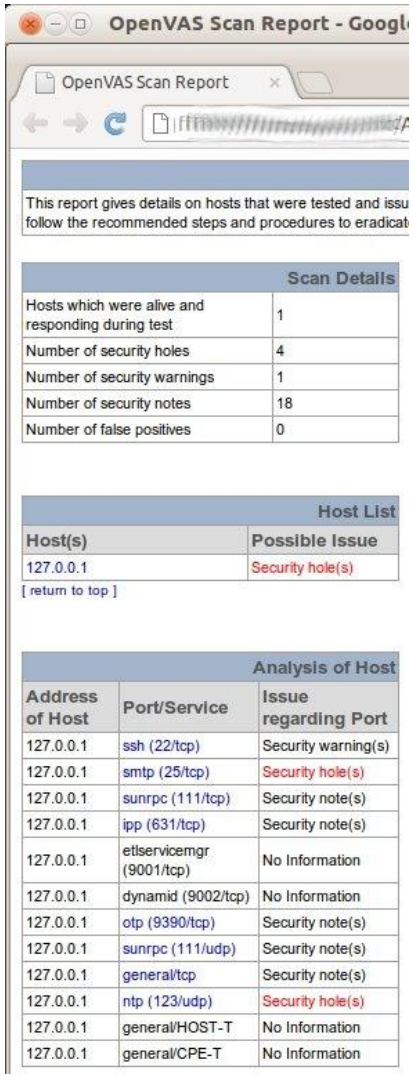
ÓBUDAI EGYETEM

Design and Implementation (3)

- How does the testjob work?
  1. Download the precompiled OpenVAS server, client and libs.
  2. Set up the environment (export path of binaries and libraries).
  3. Update NVTs.
  4. Start the OpenVAS server.
  5. Connect the client to the server and scan the current machine.
  6. Create a result file from the output of the vulnerability assessment.

Results of the vulnerability assessments

- Some examples:
  - results0.html
  - results1.html
  - results2.html
  - results3.html



ÓBUDAI EGYETEM

Conclusion and future work

- AVAT can cover broad range of security issues (based on OpenVAS)
- AVAT can provide unified vulnerability assessment for multi-middleware DCIs (Transparent interoperability issue resolution)
- Our solution is fully compliant with the available DCI solutions (ARC and gLite so far) extension is „easy”
- Service-like DCI vulnerability assessment is feasible and could help to create more secure infrastructure (support system administrators)
- Sampling (statistical selection) can be effectively used for DCI vulnerability assessments
- Vulnerability assessment was done on some parts of the HP-SEE and SEE-GRID-SCI infrastructure
  - HP-SEE: investigated 4 HPC Centers, 1 center has 4 vulnerabilities (total 4, avg. 1/center)
  - SEEGRID: investigated 16 sites, 12 sites has vulnerabilities (total 52, avg. 3.25/site)
  - Most of the issues can be fixed with updates.

ÓBUDAI EGYETEM

Thank you for the attention!

Questions?