

# *Security mechanisms in the Grid Appliance*

Grid Appliance Development Team  
ACIS Laboratory  
<http://wow.acis.ufl.edu>

# Summary

---

- Target audience
  - Users of the Grid appliance
- Material covered
  - Overview
  - Host sandboxing – VMs
  - Virtual private network – IPsec
  - Firewalling
  - Host-only networking
  - Miscellaneous

# Target users

---

- Please refer first to the Grid appliance introductory documentation for a discussion of Grid appliance use cases
- This presentation details the security capabilities integrated in the Grid appliance
  - We assume you are familiar with basics of the Grid appliance, Linux
  - Familiarity with security concepts such as public-key based authentication and privacy and firewalling will help you follow the discussion

# Overview

---

- Security is complex
  - There are many aspects to it
  - Security systems can be complex to deploy
- Grid Appliance overall approach
  - Provide security at a level our target users are expected to be comfortable
  - Virtualization is key:
    - Sandboxing
    - Reuse existing, well-tested security software – unmodified
    - Homogeneous environment simplifies configuration

# Our environment and goals

---

- The Grid Appliance allows you to run applications on remote resources
  - And remote users to run applications on your resource
- Desirable guarantees:
  - Remote access must always be through job scheduler
  - Jobs must run with low privilege
  - Communication must be confined to virtual private network
  - Must be able to strongly authenticate appliance users
  - Must be able to encrypt all inter-appliance communication

# Remote access

---

- Leverage Condor
  - Appliance does not run services allowing remote login through the IPOP network
    - Exception: secure shell (SSH) with public key authentication may be enabled for remote administration
- Only by appliances which have been registered with a certificate authority
  - IPsec-based authentication at the IP layer

# Privilege level of jobs

---

- Leverage Linux protection mechanisms
- Jobs run as unprivileged user “nobody”
  - Handled by Condor
- Jobs run within a virtual machine
  - Jobs do not have access to resources of your host
    - Even if they crash or wipe out the VM’s disk, the rest of your computer is isolated
  - Easy to shut down, suspend
  - Possible to limit available VM resources to avoid denial of service

# Traffic confinement

---

- Leverage IPtables firewalls
- Prevent distributed denial of service (DDoS) attacks to Internet hosts
- Firewall built-in and pre-configured:
  - Only allows traffic from unprivileged users to go to private IPOP addresses of other appliances – not to public Internet hosts
  - “Root” user is not constrained
  - Local firewall rules can be applied to further restrict or relax constraints depending on user needs (e.g. dropping rules in a pool already confined to a LAN)

# Strong authentication

---

- Leverage IPsec and public-key infrastructure (PKI)
  - Technology widely used in VPNs
  - Kernel support and user-level “racoon” utility available and used in various Unix systems (Linux, BSD, Mac OS/X)
- Appliances can only send IP packets to other appliances after receives a certificate signed by an authority (CA) for its virtual IP

# Privacy

---

- Again, leverage IPsec
- Strong cryptographic encryption for all IP packets exchanged among appliances
  - E.g. Condor files transmitted during job submission

# Security configuration

---

- VMs + virtual network greatly facilitates configuration of security mechanisms
  - All nodes have the same software
    - Simpler to keep up to date with security patches, install and properly configure Condor, IPsec, IPtables
  - All nodes within the same network mask
    - Simple configuration of IPsec: ~20 lines of config. files to enable point-to-point IPsec for entire class-A network

# Host-only networking

---

- For usability reasons, need simple mechanisms for the owner of the appliance to access it
- Leverage VM host-only interfaces to provide streamlined access to desktop end users
  - Assume desktop owner is the appliance user
  - Local-host SAMBA exported folder for simple drag-and-drop file sharing
  - Local-host SSH/SCP server

# Miscellaneous

---

- Use stable Debian release as a basis for appliance
- Run only strictly necessary services
- Appliances check nightly (and on boot) for updates
  - E.g. security patches, Condor upgrades
  - IPsec certificate revocation lists can be pushed during an update

# Credits and acknowledgments

---

- Development team at the ACIS Lab, U. Florida
  - <http://wow.acis.ufl.edu>
- CI-TEAM at U. Florida
  - <http://ci-team.acis.ufl.edu>

# Sponsors

---

- National Science Foundation
  - Under grants SCI-0537455, SCI-0438246, ANI-0300118



Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.