SELINIX IN PRODUCTION
Deploying SELinux successfully in production environments
Jerone, Lukas, Daniel
Tuesday, May 8 10:30 AM - 11:15 AM
Agenda

- Enabling customers
- SELinux FAQ
- Containers way
- Panel discussion
ENABLING CUSTOMERS
In Production downtime is costly and change can come slowly. Uptime and functionality are the main driving forces.
Isn't SELinux old? Who is asking for it now?

- Seeking add additional protection through application containment.
  - Financial sector customers

- Seeking to increase overall security profile of their environment.
  - Energy and Utility sector customers
  - Financial sector customers
  - Public sector customers and contractors
    - Defense Information System Agency (DISA) has mandated in the Secure Technical Implementation Guide (STIG) that SELinux must be in enforcing mode if you are running RHEL 6 or RHEL 7.

- These initiatives are being done at scale for the entire environment.
Enabling SELinux at scale is difficult and timely

- Must ensure applications remain functional.
- Building internal experts:
  - Administrators
    - Focused on application deployment and system administration.
  - Developers
    - Development of SELinux policies for custom applications.
- Building internal resolution strategies for SELinux application functionality issues.
- Ensuring all administrative staff is trained in handling a SELinux enabled environment.
- Facility to analyse and monitor of audit logs for a large amount of systems once deployed.
- Deploying SELinux slowly is a key factor to its success in a production environment.
Summary of enablement steps for Admins

1. **Turn on SELinux in Permissive mode**
   - Permissive is a debug mode. It does not enforce policies, only reports.
   - Ensure auditd is enabled, and at least 2 - 3 GB is space available in /var/log/audit/
   - Requires a reboot if SELinux is disabled
     - **i.** THIS CAN BE VERY TIMELY IN PRODUCTION
     - **ii.** Ensure to turn on filesystem relabel on reboot

2. **Observation**
   - Observe audit logs for SELINUX avc denial messages that could be preventing functionality.

3. **Remediation**
   - This is a critical thinking exercise. How to properly remediate SELINUX denials.

4. **Apply Remediation**

5. **Repeat steps 2 - 4 for a given amount of time**
   - While doing so create runbook or guide for SELinux enablement of Operating System or Application.

6. **Set SELinux in Enforcing Mode**
Application Deployment Workflow for Admins

● DEV / Test environment to vet SELinux issues before Production deployment is critical.
  ○ Simulate how functionally use application in Production.
  ○ If possible running test suites to exercise application functionality can greatly help also.

● Do this process per application.

● As remediate SELinux issues, make remediations as apart of the individual application's deployment package or instructions.

● On initial rollout into Production, leave system in Permissive mode for a given period of time to ensure there are no functional issues.

● Enable Enforcing mode for application in Production.
Remediation of SELinux denials

● Remediation is a critical thinking problem.
  ○ May require knowledge of how the application is supposed to work.

● Red Hat provides great tools like `audit2allow` and `audit2why` that extremely helpful with resolving SELinux issues, though you can't always take what they say as the proper solution.

● Example SELINUX audit log denial:

```
type=AVC msg=audit(1511797905.636:50): avc: denied { open } for pid=2708
  comm="rsyslogd" path="/etc/rsyslog.d/test2.conf" dev="dm-1" ino=1308304
  scontext=system_u:system_r:syslogd_t:s0 tcontext=unconfined_u:object_r:user_tmp_t:s0
tclass=file
```
Remediation of SELinux denials

- Output from audit2allow:

```
#========== syslogd_t ============
allow syslogd_t user_tmp_t:file open;
```

- In this example the tool will not give the correct solution.
  - The tool is saying to create a custom selinux policy module to grant open access from domain syslogd_t to any file of type user_tmp_t.
    - Meaning any process running as syslogd_t can open any file of type user_tmp_t.

- The better solution is that file `/etc/rsyslog.d/test2.conf` should be of type syslog_conf_t and not user_tmp_t.
  - With this we only relabel the file type syslog_conf_t.
Administrative tools

- Remote log gathering, analysis, and monitoring tools
  - Help monitor environment and catch SELinux activity.
  - Example Tools:
    - Splunk
      - Linux Auditd app
    - Nagios

- Configuration Management tools
  - Help with SELinux remediation deployment and application deployment.
  - Example Tools:
    - Ansible / Ansible Tower
    - Puppet
Custom & Vendor applications

- SELinux policies may want to be created for in-house custom applications:
  - In-house developers may need training, though by creating SELinux policies for custom applications:
    - Save administrators time, with less SELinux remediation since applications will have their own policies.
    - Developers will also take on SELinux testing of the custom application before it goes out for deployment. As it will become apart of their development testing and workflow.
  - Red Hat can help with mentoring and policy development with in-house developers.

- If a vendor application does not provide an SELinux policy:
  - First contact the vendor and see if they have an SELinux policy for their application.
  - Contact Red Hat and find out if that vendor is a partner. Red Hat maybe able to help.
  - If you decide to create an SELinux policy for the application, try and get blessing from the vendor to avoid any support issues in the future from the vendor.

- Majority of Red Hat supplied applications come with a supplied SELinux policy.
If you need help contact Red Hat
SELINUX FAQ
What kind of security does SELinux provide for my production environment?
Proactive security

Window of vulnerability is filled by proactive security
Could SELinux mitigate damage caused by Meltdown and Spectre?
Meltdown & Spectre vs. SELinux

Unfortunately SELinux cannot mitigate damage caused by recently disclosed vulnerabilities Meltdown and Spectre.
Is there an exploit example where SELinux help to protect your system?
http://y2u.be/Ysshrh4aG0s
Is it possible to deploy SELinux configuration to production environment?
Yes, it’s possible using Ansible!
Permissive mode = Debugging mode
Accesses are logged
Not enforced
Enforcing

SELinux security policy is enforced by kernel
Ansible Galaxy provides Linux-system-roles. SELinux role

Essentially provide mechanisms to manage local customizations:

- Set enforcing/permissive
- restorecon portions of filesystem tree
- Set/Get Booleans
- Set/Get file contexts
- Manage logins
- Manage ports
What is a key to understand SELinux?
SELinux policy rules
Describe an **Interaction** between processes and system resources
SELinux `Allow` rule syntax with `Types`
allow type1 type2:object_class
permission;
allow apache_t apache_log_t:file read;
apache_process apache_log

are labels
Assigned to processes
Assigned to system resources
by selinux security policy
map real system entities into the SELinux world
SELinux keeps your container in its own space
SELinux user:SELinux role:SELinux type:SELinux category

- system_u:object_r:container_t:c306,c536
- system_u:object_r:container_t:c206,c636
- system_u:object_r:container_t:c406,c736
Containers redesign the way Linux Works

Time to rethink the OS, and SELinux

- SELinux is about controlling what a group of processes can do on a system.
Containers redesign the way Linux Works
Time to rethink the OS, and SELinux

- SELinux is about controlling what a group of processes can do on a system.
  - OpenShift V2.0
Containers redesign the way Linux Works

Time to rethink the OS, and SELinux

- SELinux is about controlling what a group of process can do on a system.
  - OpenShift V2.0
  - ps command or ls -l /dev
Containers redesign the way Linux Works

Time to rethink the OS, and SELinux

- SELinux is about controlling what a group of processes can do on a system.
  - OpenShift V2.0
  - `ps` command or `ls -1 /dev`
- Containers are about controlling what a group of processes can do on a system
Containers redesign the way Linux Works

Time to rethink the OS, and SELinux

- SELinux is about controlling what a group of process can do on a system.
  - OpenShift V2.0
  - ps command or ls -l /dev

- Containers are about controlling what a group of processes can do on a system
  - Only allow the writable content into the container
Containers redesign the way Linux Works
Time to rethink the OS, and SELinux

- SELinux is about controlling what a group of process can do on a system.
  - OpenShift V2.0
  - ps command or ls -l /dev
- Containers are about controlling what a group of processes can do on a system
  - Only allow the writable content into the container
  - Remove all content that you don’t want the container processes out of the container.
Containers redesign the way Linux Works

Time to rethink the OS, and SELinux

- SELinux is about controlling what a group of process can do on a system.
  - OpenShift V2.0
  - ps command or ls -l /dev
- Containers are about controlling what a group of processes can do on a system
  - Only allow the writable content into the container
  - Remove all content that you don’t want the container processes out of the container.
  - What happens in Vegas stays in Vegas.
Containers redesign the way Linux Works
Time to rethink the OS, and SELinux

- SELinux is about controlling what a group of processes can do on a system.
  - OpenShift V2.0
  - ps command or ls -l /dev

- Containers are about controlling what a group of processes can do on a system
  - Only allow the writable content into the container
  - Remove all content that you don’t want the container processes out of the container.
  - What happens in Vegas stays in Vegas.

- SE-Android
Containers redesign the way Linux Works
Time to rethink the OS, and SELinux

- SELinux is about controlling what a group of processes can do on a system.
  - OpenShift V2.0
  - ps command or ls -l /dev
- Containers are about controlling what a group of processes can do on a system
  - Only allow the writable content into the container
  - Remove all content that you don’t want the container processes out of the container.
  - What happens in Vegas stays in Vegas.
- SE-Android
- Docker Exploits
DISCUSSION PANEL
SELINUX COLORING BOOK

Don’t miss these labs & sessions coming up this week

**MAY 8**
1:00 - 3:00 PM
Defend yourself using built-in Red Hat Enterprise Linux security technologies
Session code: L1036

**MAY 9**
4:00 - 6:00 PM
A practical introduction to container security
Session code: L1007

**MAY 10**
1:00 - 1:45 PM
Security-Enhanced Linux for mere mortals
Session code: S1931

1:45 - 3:45 PM
A practical introduction to container security
Session code: L1007R
THANK YOU

plus.google.com/+RedHat
linkedin.com/company/red-hat
youtube.com/user/RedHatVideos
facebook.com/redhatinc
twitter.com/RedHatNews