Security Compliance with OpenSCAP

Automatically find vulnerabilities and configuration issues of your infrastructure

Martin Preisler
Software Engineer, Red Hat, Inc.
AGENDA

- **software flaws - vulnerabilities**
  - checking machines for vulnerabilities
  - checking containers for vulnerabilities

- **configuration flaws - weaknesses**
  - what is a security policy
  - SCAP introduction
  - security compliance for a single machine
  - security policy customization
  - remediation
  - using SCAP with Red Hat Satellite 6

- **future plans**

There will be demos!
AGENDA

Security is a very broad topic. In this session we will be discussing:

- software flaws - vulnerabilities
- configuration flaws - weaknesses
VULNERABILITY
what is a software vulnerability...

- a weakness that can be exploited by a threat
- a weakness in the software that allows attacker to reduce information assurance
- can lead to compromise of security
VULNERABILITIES

Undiscovered vulnerabilities are bad.

- But not all that bad, everybody has them
- It's a lot of effort to use those for exploits
- Mitigate with SELinux
VULNERABILITIES

Known vulnerabilities are *much worse.*

- CVE-2016-1283
- Details are publicly available
VULNERABILITIES

*Known* vulnerabilities are sometimes so bad that they have *fancy names*!

- Shellshock, POODLE, VENOM, ...
VULNERABILITIES
... and sometimes even logos!

Known vulnerabilities:

- assigned CVEs - CVE-2014-0160
- details are public for everyone
- ready-made exploits may be available
VULNERABILITIES
Not all vulnerabilities are equal.

Let’s prioritize:

● vulnerabilities are dangerous
● there is not much we can do about the undiscovered ones
● let’s never have any known ones in our infrastructure!
USE-CASE 1: AUTOMATICALLY CHECK VULNERABILITIES
VULNERABILITY ASSESSMENT ON RHEL 6
Let’s discuss how to scan a single Red Hat Enterprise Linux 6 machine.

There are three steps to perform:

1. Download the CVE data
2. Execute the oscap tool
3. Review the results
COMMANDS TO SCAN RHEL 6 FOR CVEs

# cd /tmp
# wget https://www.redhat.com/security/data/oval/Red_Hat_Enterprise_Linux_6.xml
# oscap oval eval --results /tmp/results.xml --report /tmp/report.html
    Red_Hat_Enterprise_Linux_6.xml
# firefox /tmp/report.html
VULNERABILITY SCAN RESULTS

After the command is invoked this is what we can see in stdout.

```
Definition oval:com.redhat.rhsa:def:20151682: false
Definition oval:com.redhat.rhsa:def:20151668: false
Definition oval:com.redhat.rhsa:def:20151643: false
Definition oval:com.redhat.rhsa:def:20151640: false
Definition oval:com.redhat.rhsa:def:20151636: false
Definition oval:com.redhat.rhsa:def:20151634: false
Definition oval:com.redhat.rhsa:def:20151633: false
Definition oval:com.redhat.rhsa:def:20151623: true
Definition oval:com.redhat.rhsa:def:20151603: false
Definition oval:com.redhat.rhsa:def:20151586: false
Definition oval:com.redhat.rhsa:def:20151581: false
Definition oval:com.redhat.rhsa:def:20151544: false
Definition oval:com.redhat.rhsa:def:20151526: false
Definition oval:com.redhat.rhsa:def:20151513: false
Definition oval:com.redhat.rhsa:def:20151499: false
Definition oval:com.redhat.rhsa:def:20151486: false
Definition oval:com.redhat.rhsa:def:20151485: false
Definition oval:com.redhat.rhsa:def:20151482: false
Definition oval:com.redhat.rhsa:def:20151471: false
Definition oval:com.redhat.rhsa:def:20151462: false
Definition oval:com.redhat.rhsa:def:20151466: false
Definition oval:com.redhat.rhsa:def:20151459: false
```
VULNERABILITY SCAN RESULTS

After the command is invoked this is what we can see in stdout.
## Vulnerability Scan Results

Let's see more details by opening the HTML report.

<table>
<thead>
<tr>
<th>ID</th>
<th>Result</th>
<th>Class</th>
<th>Reference ID</th>
<th>Title</th>
</tr>
</thead>
</table>
VULNERABILITY SCAN RESULTS
After installing system updates and rebooting the vulnerability is gone.

<table>
<thead>
<tr>
<th>Component</th>
<th>State</th>
<th>Patch</th>
<th>Details</th>
</tr>
</thead>
</table>
DEMO: FIND VULNERABILITIES ON RED HAT ENTERPRISE LINUX 7
WHAT ABOUT CONTAINERS?

Scanning containers one by one like this is impractical...

Production deployments are increasingly using containers. This brings new challenges.

- installing the oskap tool in every container is impractical
- single-purpose containers ➔ many different containers and images
ATOMIC SCAN
New feature in Atomic 1.4

Scan containers and container images for CVEs.

```
root@t440s ~ # atomic scan 6c3a84d798dc
Container/Image     Cri Imp Med Low
-------------------- --- --- --- ---
6c3a84d798dc        0    0   4    0
```
ATOMIC SCAN detailed
--detail prints out the errata and CVE details and references

```
root@t440s ~ # atomic scan --detail 6c3a84d798dc

6c3a84d798dc
OS : Red Hat Enterprise Linux Server release 7.2 (Maipo)
Moderate : 4

cve : RHSA-2016:0008: openssl security update (Moderate)
cve url : https://access.redhat.com/security/cve/CVE-2015-7575
rhsa id : RHSA-2016:0008-00
rhsa url : https://rhn.redhat.com/errata/RHSA-2016-0008.html

cve : RHSA-2016:0007: nss security update (Moderate)
cve url : https://access.redhat.com/security/cve/CVE-2015-7575
rhsa id : RHSA-2016:0007-00
rhsa url : https://rhn.redhat.com/errata/RHSA-2016-0007.html

cve : RHSA-2015:2617: openssl security update (Moderate)
cve url : https://access.redhat.com/security/cve/CVE-2015-3194
rhsa id : RHSA-2015:2617-00

cve : RHSA-2015:2550: libxml2 security update (Moderate)
cve url : https://access.redhat.com/security/cve/CVE-2015-1819
rhsa id : RHSA-2015:2550-01
```
ATOMIC SCAN WITH MULTIPLE TARGETS
Scan all your containers and container images with a single command.

Three options are available, scan all containers, scan all images and scan both.

- atomic scan --containers
- atomic scan --images
- atomic scan --all
DEMO: ATOMIC SCAN ON RED HAT ENTERPRISE LINUX 7
HOW DOES ATOMIC SCAN WORK?
we can’t trust what we don’t understand...

**DETECT OS VERSION**
Different operating systems have different CVEs.

**DOWNLOAD CVE FEED**
Based on the OS version we download CVE feed from the vendor.

**RUN OSCAP TOOL**
OpenSCAP compares installed versions with version ranges in the CVE feed.
FOCUS OF THIS SESSION

Security is a very broad topic. In this session we will be discussing:

- software flaws - vulnerabilities
- configuration flaws - weaknesses
SECURITY POLICY
what it means to secure a system

Usually in text form or a PDF. Security policy contains a set of rules, each rule has:

● description
● rationale
● how to check
● how to fix
## SECURITY POLICY EXAMPLE

*excerpt from PCI-DSS*

<table>
<thead>
<tr>
<th>PCI DSS Requirements</th>
<th>Testing Procedures</th>
<th>Guidance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1.5 Description of groups, roles, and responsibilities for management of network components</td>
<td><strong>1.1.5.a</strong> Verify that firewall and router configuration standards include a description of groups, roles, and responsibilities for management of network components.</td>
<td>This description of roles and assignment of responsibilities ensures that personnel are aware of who is responsible for the security of all network components, and that those assigned to manage components are aware of their responsibilities. If roles and responsibilities are not formally assigned, devices could be left unmanaged.</td>
</tr>
<tr>
<td></td>
<td><strong>1.1.5.b</strong> Interview personnel responsible for management of network components to confirm that roles and responsibilities are assigned as documented.</td>
<td></td>
</tr>
<tr>
<td>1.1.6 Documentation and business justification for use of all services, protocols, and ports allowed, including documentation of security features implemented for those protocols considered to be insecure.</td>
<td><strong>1.1.6.a</strong> Verify that firewall and router configuration standards include a documented list of all services, protocols and ports, including business justification for each—for example, hypertext transfer protocol (HTTP) and Secure Sockets Layer (SSL), Secure Shell (SSH), and Virtual Private Network (VPN) protocols.</td>
<td>Compromises often happen due to unused or insecure service and ports, since these often have known vulnerabilities and many organizations don’t patch vulnerabilities for the services, protocols, and ports they don’t use (even though the vulnerabilities are still present). By clearly defining and documenting the services, protocols, and ports that are necessary for business, organizations can ensure that all other services, protocols, and ports are disabled or removed.</td>
</tr>
<tr>
<td></td>
<td><strong>1.1.6.b</strong> Identify insecure services, protocols, and ports allowed; and verify that security features are documented for each service.</td>
<td>If insecure services, protocols, or ports are necessary for business, the risk posed by use of these protocols should be clearly understood and accepted by the organization, the use of the protocol should be justified, and the security features that allow these protocols to be used securely should be documented and implemented. If these insecure services, protocols, or ports are not necessary for business, they should be disabled or removed.</td>
</tr>
<tr>
<td></td>
<td><strong>1.1.6.c</strong> Examine firewall and router configurations to verify that the documented security features are implemented for each insecure service, protocol, and port.</td>
<td></td>
</tr>
</tbody>
</table>
AUTOMATING A SECURITY POLICY
for non-interactive machine processing

- very long bash script
- set of scripts with some harness running them
- proprietary solution
- SCAP
WHAT IS SCAP?

a way to express security policies in machine readable form.

SCAP is a NIST standard. It contains a set of data formats for security policies.

- rule metadata - description, rationale, identifiers
- automatic compliance checking
- automatic fixing

SCAP uses other technologies such as XCCDF, OVAL, CPE, CVE and OCIL.
ADVANTAGES OF SCAP
standards help avoid lock-in

- royalty free
- multiple implementations
- can mix and match scanner tools and security policies
- deploy a heterogeneous mix of tools from different vendors
SCAP SECURITY POLICY EXAMPLE

HTML guide generated from SCAP security policy

Network Configuration and Firewalls
Most machines must be connected to a network of some sort, and this brings with it the substantial risk of network attack. This section discusses the security impact of decisions about networking which must be made when configuring a system.

This section also discusses firewalls, network access controls, and other network security frameworks, which allow system-level rules to be written that can limit an attackers' ability to connect to your system. These rules can specify that network traffic should be allowed or denied from certain IP addresses, hosts, and networks. The rules can also specify which of the systems network services are available to particular hosts or networks.

IPSec Support
Support for Internet Protocol Security (IPsec) is provided in Red Hat Enterprise Linux 7 with Libreswan.

Install libreswan Package
The Libreswan package provides an implementation of IPsec and IKE, which permits the creation of secure tunnels over untrusted networks. The libreswan package can be installed with the following command:

```
$ sudo yum install libreswan
```

Automated
Providing the ability for remote users or systems to initiate a secure VPN connection protects information when it is transmitted over a wide area network.

- **Identifiers**: CCE-RHEL7-CCE-TBD
- **References**: AC-17, MA-4, SC-9, 1130, 1131, Req-4
- **Remediation script**: yum -y install libreswan
TWO TYPES OF SCAP SECURITY POLICIES

VULNERABILITY ASSESSMENT
- detect CVEs
- Heartbleed
- Shellshock
- Ghost
- VENOM
- ...

SECURITY COMPLIANCE
- proper configuration
- hardening
- USGCB
- PCI-DSS
- DISA STIG
- ...

TWO SCAP USE-CASES

VULNERABILITY ASSESSMENT

are my machines vulnerable to:
Heartbleed?
Shellshock?
Ghost?
VENOM?
...

SECURITY COMPLIANCE

is root login over ssh forbidden?
is SELinux enabled and enforcing?
are we using strict password policy?
are obsolete / insecure services disabled?
...

USE-CASE 2:
SECURITY COMPLIANCE
FOR A SINGLE MACHINE
OPENSCAP
open-source SCAP 1.2 implementation

- certified by NIST since 2014
- library and a command-line interface
- GUI frontend is available - SCAP Workbench
SCAP SECURITY GUIDE
open-source SCAP security policy project

- community project
- content for multiple products - Red Hat Enterprise Linux, Fedora, CentOS, Firefox, ...
- multiple policies for each product - USGCB, PCI-DSS, DISA STIG, ...
SCANNING A SINGLE MACHINE
let’s set-up a Red Hat Enterprise Linux 7.2 machine as close to PCI-DSS as possible

We will need the following to perform a PCI-DSS scan:

- Red Hat Enterprise Linux 7.2
- OpenSCAP and SCAP Workbench
- PCI-DSS from SCAP Security Guide
INSTALL THE NECESSARY TOOLS
(assuming Red Hat Enterprise Linux 7.2)

```
# yum install scap-security-guide
# yum install scap-workbench
```
After starting SCAP Workbench we will be asked to select the security policy we want to load.

Let’s select ssg-rhel7-ds.xml, which is a security policy for Red Hat Enterprise Linux 7 in the datastream SCAP format.
INITIAL SCAN
let’s do a quick scan to establish a baseline

1. select the PCI-DSS profile
2. keep local machine selected
3. click Scan
INITIAL SCAN
let’s do a quick scan to establish a baseline

1. select the PCI-DSS profile
2. keep local machine selected
3. click Scan
INITIAL RESULTS

Compliance and Scoring

The target system did not satisfy the conditions of 43 rules! Please review rule results and consider applying remediation.

Rule results

- 31 passed
- 43 failed

Severity of failed rules

- 33 low
- 9 medium

Score

<table>
<thead>
<tr>
<th>Scoring system</th>
<th>Score</th>
<th>Maximum</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>urn:xccdf:scoring:default</td>
<td>65.168396</td>
<td>100.000000</td>
<td>65.17%</td>
</tr>
</tbody>
</table>
## INITIAL RESULTS

- **Configure Syslog**

- **System Accounting with auditd** [31x fail]
  - **Configure auditd Data Retention** [3x fail]
  
<table>
<thead>
<tr>
<th>Task</th>
<th>Level</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure auditd Number of Logs Retained</td>
<td>medium</td>
<td>pass</td>
</tr>
<tr>
<td>Configure auditd Max Log File Size</td>
<td>medium</td>
<td>pass</td>
</tr>
<tr>
<td>Configure auditd max_log_file_action Upon Reaching Maximum Log Size</td>
<td>medium</td>
<td>pass</td>
</tr>
<tr>
<td>Configure auditd space_left Action on Low Disk Space</td>
<td>medium</td>
<td>fail</td>
</tr>
<tr>
<td>Configure auditd admin_space_left Action on Low Disk Space</td>
<td>medium</td>
<td>fail</td>
</tr>
<tr>
<td>Configure auditd mail_acct Action on Low Disk Space</td>
<td>medium</td>
<td>pass</td>
</tr>
<tr>
<td>Configure auditd to use audispd's syslog plugin</td>
<td>medium</td>
<td>fail</td>
</tr>
</tbody>
</table>

- **Configure auditd Rules for Comprehensive Auditing** [27x fail]

- **Records Events that Modify Date and Time Information** [5x fail]

<table>
<thead>
<tr>
<th>Task</th>
<th>Level</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record attempts to alter time through adjtimex</td>
<td>low</td>
<td>fail</td>
</tr>
<tr>
<td>Record attempts to alter time through settimeofday</td>
<td>low</td>
<td>fail</td>
</tr>
<tr>
<td>Record Attempts to Alter Time Through stime</td>
<td>low</td>
<td>fail</td>
</tr>
</tbody>
</table>
### INITIAL RESULTS

#### Set Password Maximum Age

<table>
<thead>
<tr>
<th><strong>Rule ID</strong></th>
<th>xccdf_org.ssgproject.content_rule_accounts_maximum_age_login_defs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Result</strong></td>
<td>fail</td>
</tr>
<tr>
<td><strong>Time</strong></td>
<td>2016-02-16T15:08:18</td>
</tr>
<tr>
<td><strong>Severity</strong></td>
<td>medium</td>
</tr>
<tr>
<td><strong>Identifiers and References</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Identifiers:</strong></td>
<td>CCE-27051-2</td>
</tr>
<tr>
<td><strong>References:</strong></td>
<td>IA-5(f), IA-5(g), IA-5(1)(d), 180, 199, 76, Test attestation on 20121028 by DS</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>To specify password maximum age for new accounts, edit the file <code>/etc/login.defs</code> and add or correct the following line, replacing DAYS appropriately:</td>
</tr>
<tr>
<td></td>
<td><code>PASS_MAX_DAYS DAYS</code></td>
</tr>
<tr>
<td></td>
<td>A value of 180 days is sufficient for many environments. The DoD requirement is 60.</td>
</tr>
<tr>
<td><strong>Rationale</strong></td>
<td>Setting the password maximum age ensures users are required to periodically change their passwords. This could possibly decrease the utility of a stolen password. Requiring shorter password lifetimes increases the risk of users writing down the password in a convenient location subject to physical compromise.</td>
</tr>
</tbody>
</table>
INITIAL RESULTS

**CVAL details**

**Items found violating:** The value of PASS_MAX_DAYS should be set appropriately in /etc/login.defs

<table>
<thead>
<tr>
<th>Var ref</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>oval:ssg:var:1310</td>
<td>99999</td>
</tr>
</tbody>
</table>

**Remediation script:**

```bash
var_accounts_maximum_age_login_defs="90"
grep -q ^PASS_MAX_DAYS /etc/login.defs && \
    sed -i "s/PASS_MAX_DAYS.*/PASS_MAX_DAYS $var_accounts_maximum_age_login_defs/g" /etc/login.defs
if ! [ $? -eq 0 ]; then
    echo "PASS_MAX_DAYS $var_accounts_maximum_age_login_defs" >> /etc/login.defs
fi
```
MAKING ADJUSTMENTS

Customizing "Draft PCI-DSS v3 Control Baseline for Red Hat Enterprise Linux 7 [CUSTOMIZED]"

- Restrict Root Logins
  - Direct root Logins Not Allowed
  - Restrict Virtual Console Root Logins
  - Restrict Serial Port Root Logins
  - Restrict Web Browser Use for Administrative Accounts
  - Ensure that System Accounts Do Not Run a Shell Upon Login
  - Verify Only Root Has UID 0
  - Root Path Must Be Vendor Default
- Verify Proper Storage and Existence of PasswordHashes
  - Prevent Log In to Accounts With Empty Password
  - Verify All Account Password Hashes are Shadowed
  - All GIDs referenced in /etc/passwd must be defined in /etc/
  - Verify No netrc Files Exist
- Set Password Expiration Parameters
  - minimum password length
  - maximum password age
  - minimum password age
  - warning days before password expires

Selected Item Properties

- Title: Set Password Minimum Length in login.defs
- ID: ht_rule_accounts_password_minlen_login_defns
- Type: xccdf:Rule

Description

To specify password length requirements for new accounts, edit the file /etc/login.defs and add or correct the following lines: PASS_MIN_LEN 14 The DoD requirement is 14. The FISMA requirement is 12. If a program consults /etc/login.defs and also another PAM module (such as pam_password) during a password change operation, then the most restrictive must be satisfied. See PAM section for more information about enforcing password quality requirements.

Security Identifiers

- [http://cve.mitre.org] - CCE-27123-9

Depends on Values

- minimum password length = 12
- A conditional clause for check statements. = This is a placeholder.
SAVING THE FINAL POLICY
we now have the final security policy, let’s save it for later deployment

Click File ➔ Save Customization Policy

Instead of saving the entire policy we will save the difference between stock policy and our final policy. This enables us to get improvements and bug fixes.
DEMO: CONTENT CUSTOMIZATION
AUTOMATICALLY FIXING THE ISSUES
Check Remediate to automatically fix issues after scanning.

We now have a profile defined, let’s put the machine closer to compliance. Keep this in mind when doing automatic remediation:

- remediation is potentially dangerous
- remediation cannot be undone!
REMEDIATION WITH SCAP-WORKBENCH

let’s do a quick scan to establish a baseline

- *fixed* means the remediation was successful
- some fixes require reboot
- some rules cannot be automatically fixed - these still show as *failed*
**FINAL RESULTS**

Compliance and Scoring

*There were no failed or uncertain rules. It seems that no action is necessary.*

**Rule results**

- 74 passed

**Severity of failed rules**

**Score**

<table>
<thead>
<tr>
<th>Scoring system</th>
<th>Score</th>
<th>Maximum</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>um:xccdf:scoring:default</td>
<td>65.168396</td>
<td>100.000000</td>
<td>65.17%</td>
</tr>
</tbody>
</table>
DEMO: COMMAND-LINE SCANNING OF RED HAT ENTERPRISE LINUX 7
SCANNING A CONTAINER

a command-line interface similar to oscap, scans a container “from the outside”

```
oscap-docker container $ID xccdf eval --profile xccdf_org.ssgproject.content_profile_stig-rhel7-server-upstream /usr/share/xml/scap/ssg/content/ssg-rhel7-ds.xml
```

```
oscap-docker image $ID xccdf eval --profile xccdf_org.ssgproject.content_profile_stig-rhel7-server-upstream /usr/share/xml/scap/ssg/content/ssg-rhel7-ds.xml
```
SCANNING A VIRTUAL MACHINE

a command-line interface similar to oscap, scans a VM “from the outside”

```
oscap-vm domain rhel7.2 xccdf eval --profile xccdf_org.ssgproject.content_profile_stig-rhel7-server-upstream /usr/share/xml/scap/ssg/content/ssg-rhel7-ds.xml

oscap-vm image /var/lib/libvirt/images/rhel7.2.qcow2 xccdf eval --profile xccdf_org.ssgproject.content_profile_stig-rhel7-server-upstream /usr/share/xml/scap/ssg/content/ssg-rhel7-ds.xml
```

Read more: https://martin.preisler.me/2015/10/evaluate-virtual-machines-for-scap-compliance/
USE-CASE 3: SECURITY COMPLIANCE FOR AN INFRASTRUCTURE
SCAP IN RED HAT SATELLITE 6
Red Hat Satellite 6 can be used to scan your infrastructure.

Feature highlights:
- upload SCAP content
- assign policies to hosts and hostgroups
- schedule continuous checks
- view HTML reports
SCAP IN RED HAT SATELLITE 6
upload SCAP content to create new SCAP policies
SCAP IN RED HAT SATELLITE 6
see past results

<table>
<thead>
<tr>
<th>Host</th>
<th>Date</th>
<th>Passed</th>
<th>Failed</th>
<th>Other</th>
<th>View Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>nodes.example.com</td>
<td>6 days ago</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>nodes.example.com</td>
<td>6 days ago</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
</tbody>
</table>

Displaying all entries
SCAP IN RED HAT SATELLITE 6
browse HTML report for details of a past result
SCAP IN RED HAT SATELLITE 6
Further references...

Red Hat Satellite 6.1 Feature Overview: OpenSCAP

https://www.youtube.com/watch?v=p4uNlzYld-Y
FUTURE PLANS
REWORKED ATOMIC SCAN
Faster and more robust execution, less privileges required

- Atomic does the mounting
- The OpenSCAP-daemon container does the scanning
STANDARD COMPLIANCE
Automated standard compliance checking

- Automated SCAP policy management - no need to specify content
- Not specific to the workload
- Planned for future versions of Atomic
SPECIALIZED PCI-DSS REPORT
HTML report customized for PCI-DSS compliance

- Maps PCI-DSS IDs to policy rules instead of the other way around
REMEDIATIONS FOR CONTAINERS AND VMs

Put containers and virtual machines into compliance without installing SCAP tools on them

- Change configuration of a container or container image
- Planned for future versions of Atomic
MORE SCAP POLICY OPTIONS
Continuously extend and improve the provided SCAP policies

- HIPPA
- SOX
FURTHER READING

- [https://www.open-scap.org/](https://www.open-scap.org/)
THANK YOU!

Questions?

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