Red Hat Security Seminar

Shawn D. Wells (swells@redhat.com)

Solutions Architect, Federal

Agenda

- Start: 10:30 am
- End: 1:00 pm, ish

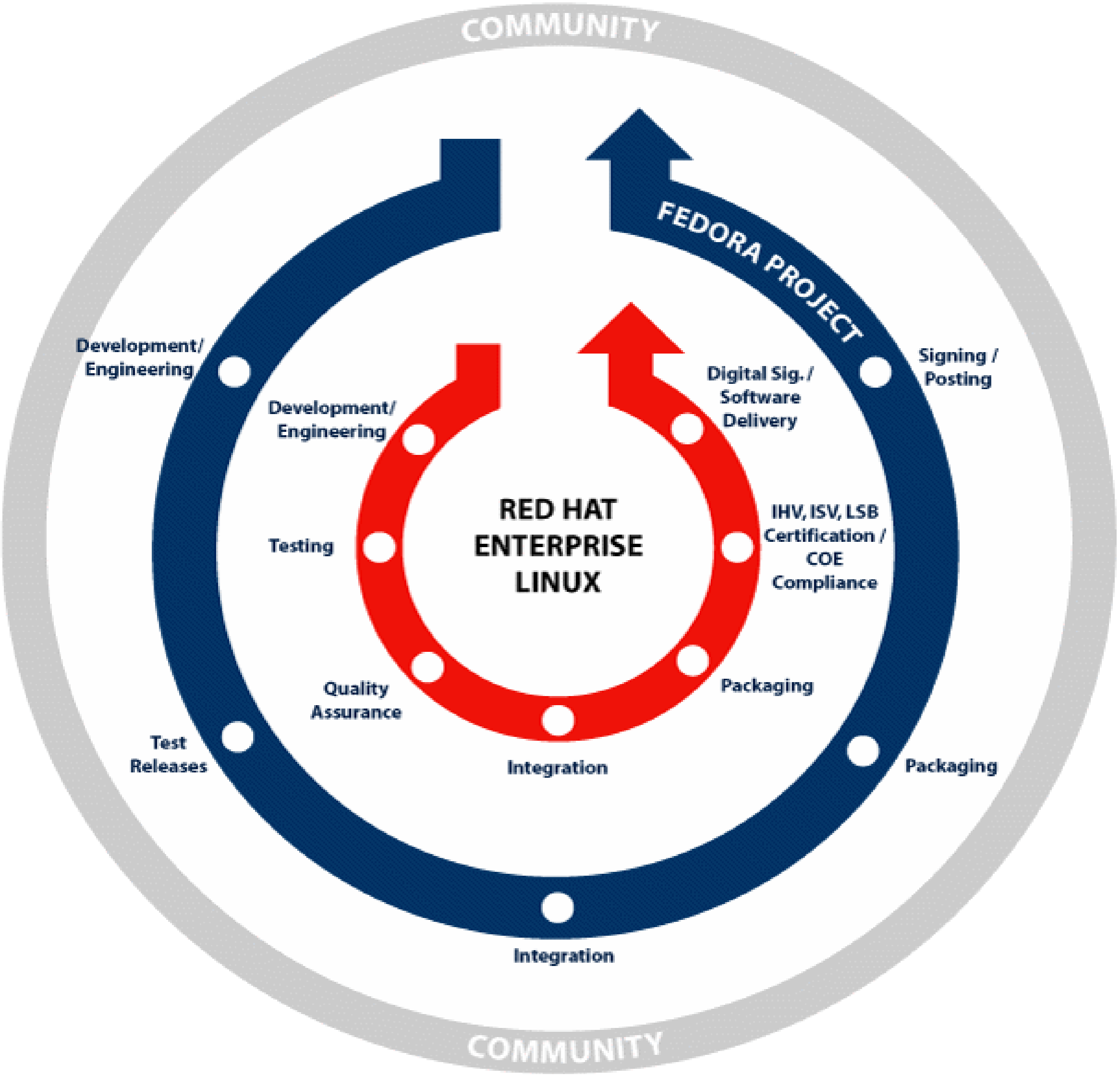
- Red Hat Emerging Technologies
  - Virtualization
  - Security/MLS/Common Criteria
  - High Availability

- Red Hat Security

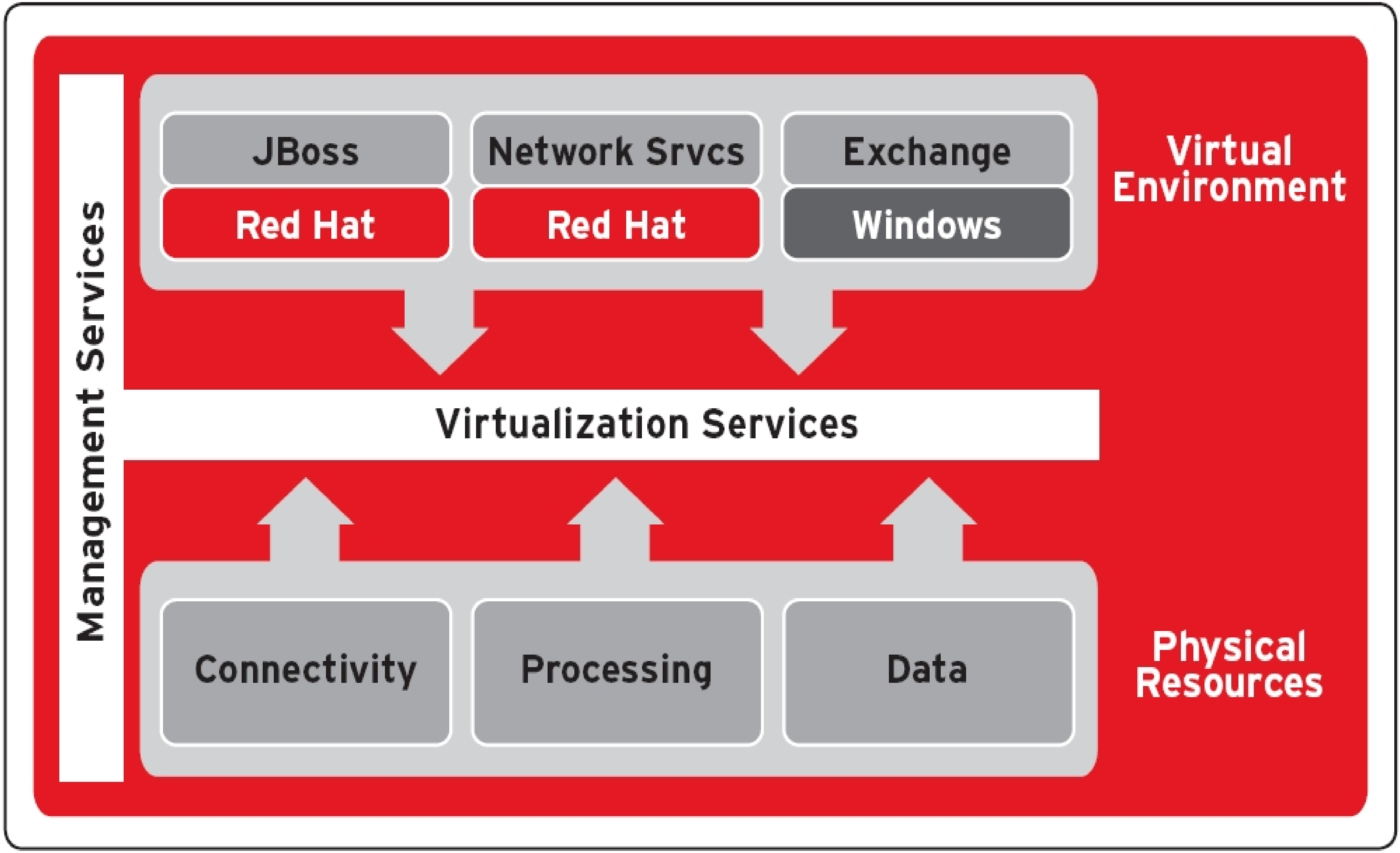
- Future Direction (Emerging Technologies)

- Summary & Close

Red Hat Development Model



Red Hat Emerging Technologies



Management Services

JBoss

Red Hat

Network Srvcs

Red Hat

Exchange

Windows

Virtual Environment

Virtualization Services

Connectivity

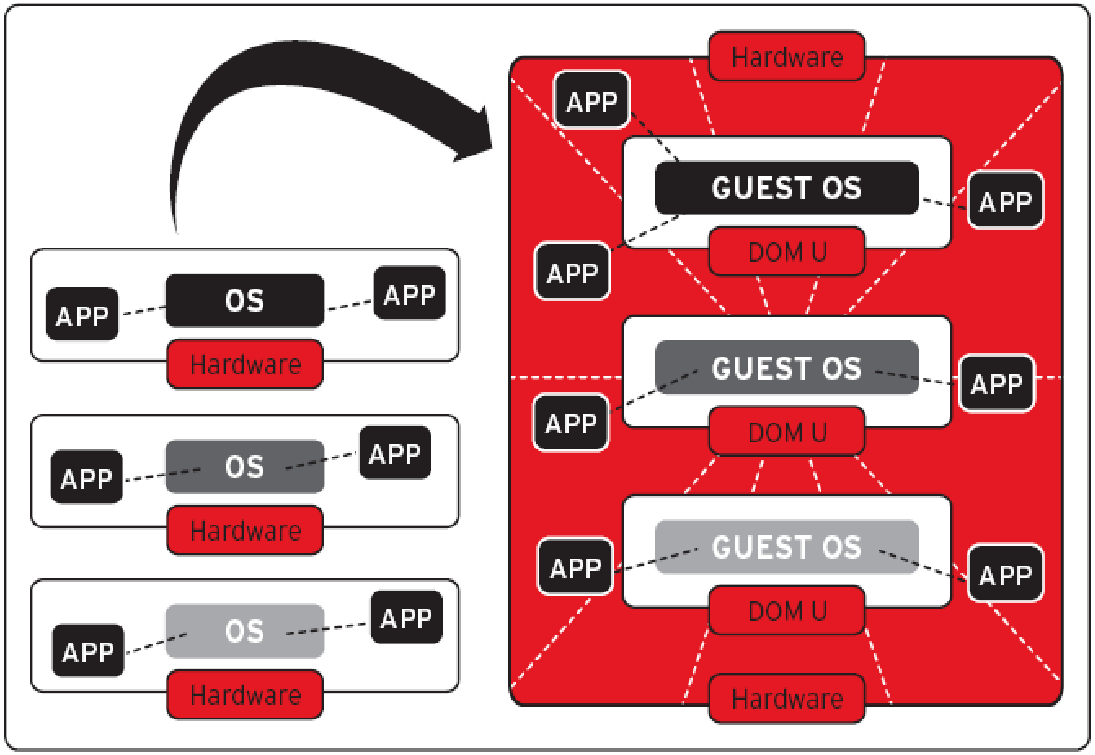
Processing

Data

Physical Resources

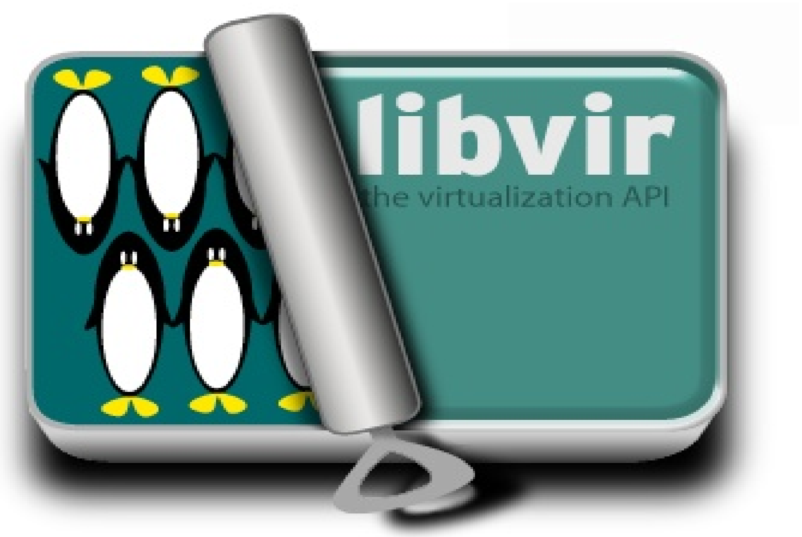
The Xen Hypervisor

- Flexible IT Services
- Disaster Tolerance
- Life Cycle Management
- Live Migration

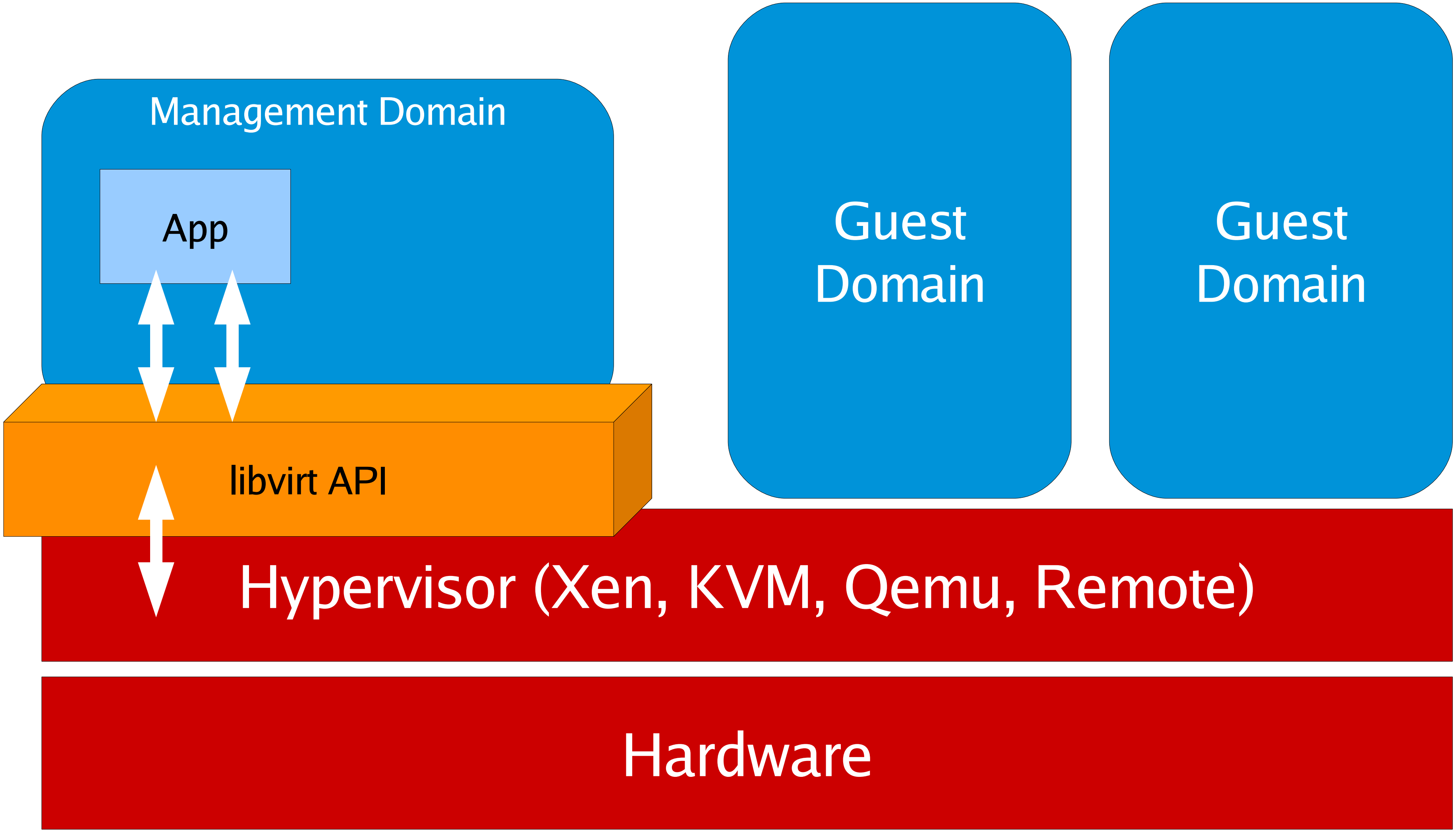


Introduction to libvirt API

- Hypervisor agnostic
- Stable API for tool/app development
  - CIM providers; Python, C bindings, scriptable
- Allows authenticated/encrypted sessions to remote hypervisors
- Current support for
  - Xen Hypervisor
  - KVM Hypervisor
  - QEMU Hypervisor

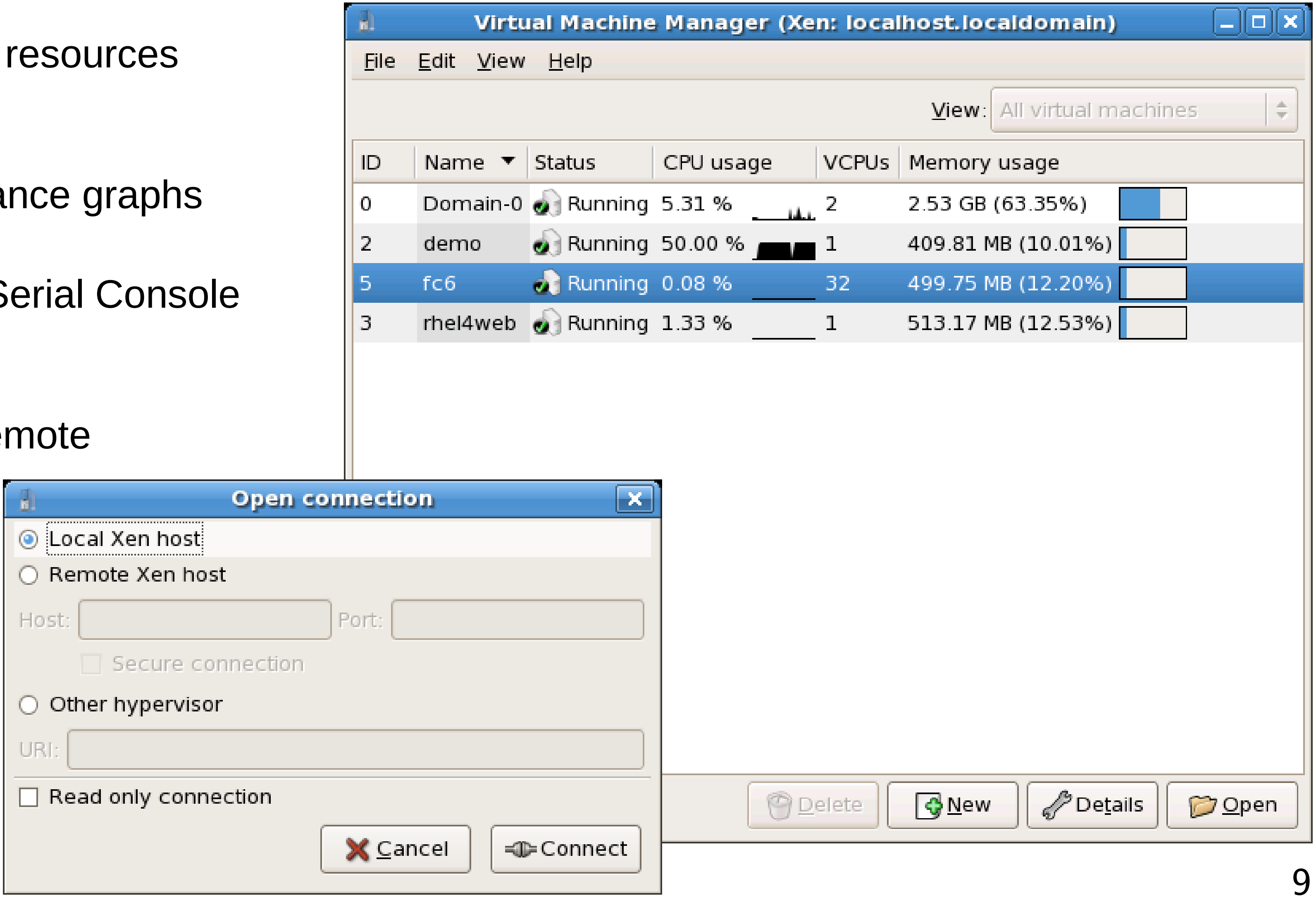


libvirt Architecture



Introduction to virt-manager

- Graphical virtual guest management
- Add/Remove resources dynamically
- Live performance graphs
- Graphical & Serial Console Emulation
- Connect to remote hosts



Virtual Machine Manager (Xen: localhost.localdomain)

ID	Name	Status	CPU usage	VCPUs	Memory usage
0	Domain-0	Running	5.31 %	2	2.53 GB (63.35%)
2	demo	Running	50.00 %	1	409.81 MB (10.01%)
5	fc6	Running	0.08 %	32	499.75 MB (12.20%)
3	rhel4web	Running	1.33 %	1	513.17 MB (12.53%)

Open connection

● Local Xen host

○ Remote Xen host

Host: Port:

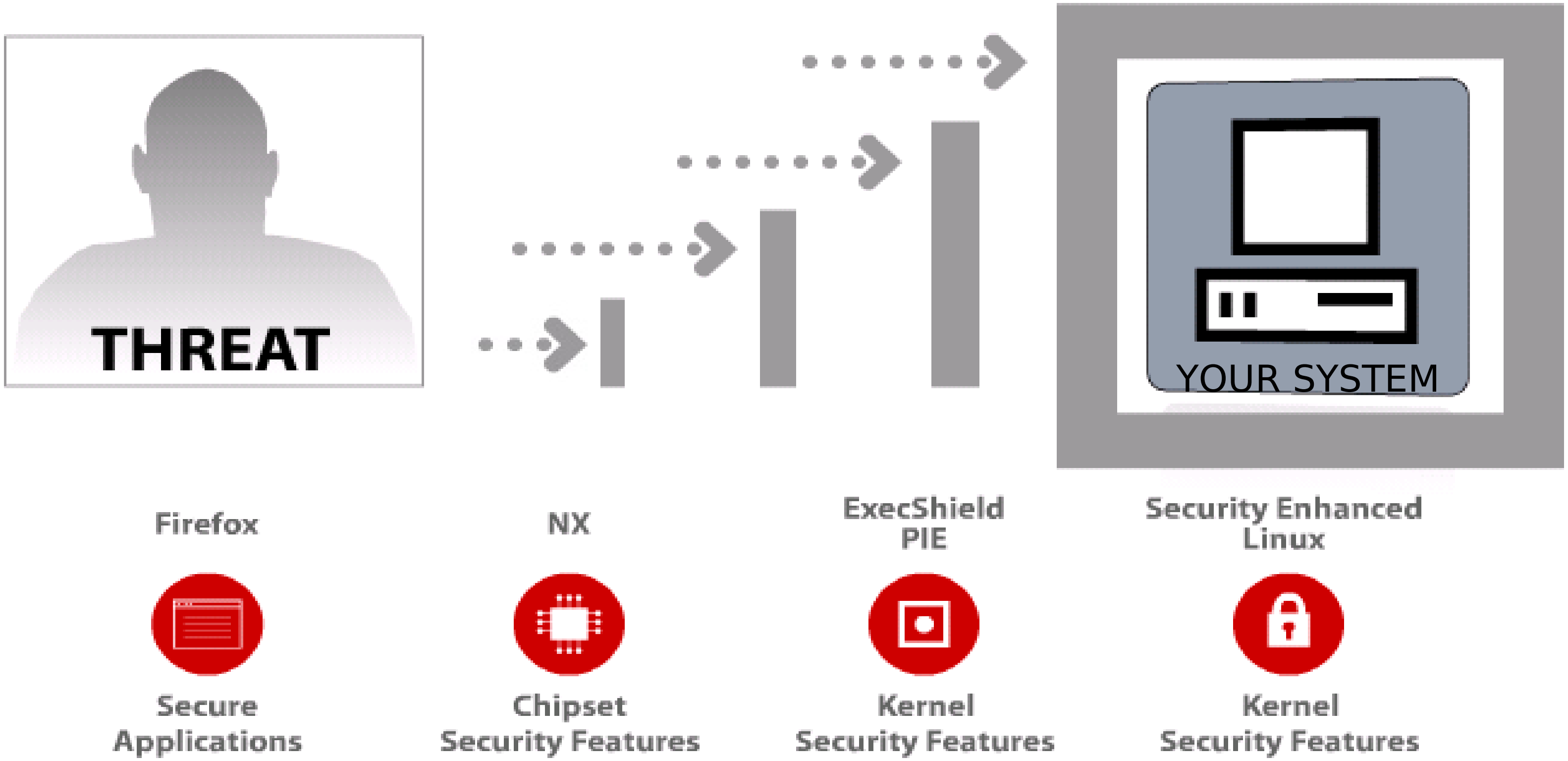
□ Secure connection

○ Other hypervisor

URI:

□ Read only connection

RHEL5 Security: A Layered Defense



THREAT

YOUR SYSTEM

Firefox

Secure Applications

NX

Chipset Security Features

ExecShield PIE

Kernel Security Features

Security Enhanced Linux

Kernel Security Features

Open Source as a Security Innovation

1. More eyes on the code, therefore less security bugs

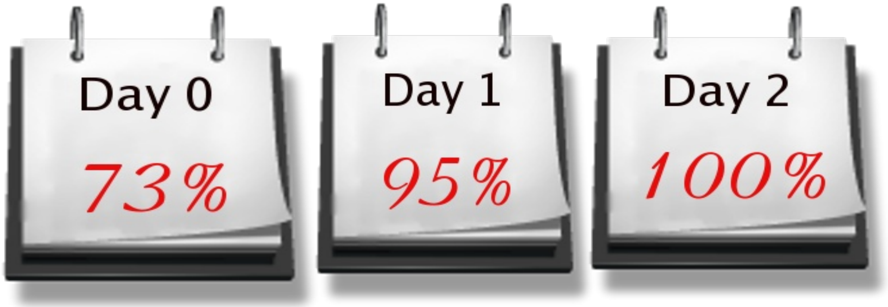
Bugs per 1000 Lines of Code

Linux 2.6 Kernel	0.17	Stanford University/Cover
Proprietary Software	10 to 20	Carnegie Mellon Cylab

Wired Magazine, Dec 2004

2. Red Hat's rapid response to any vulnerabilities

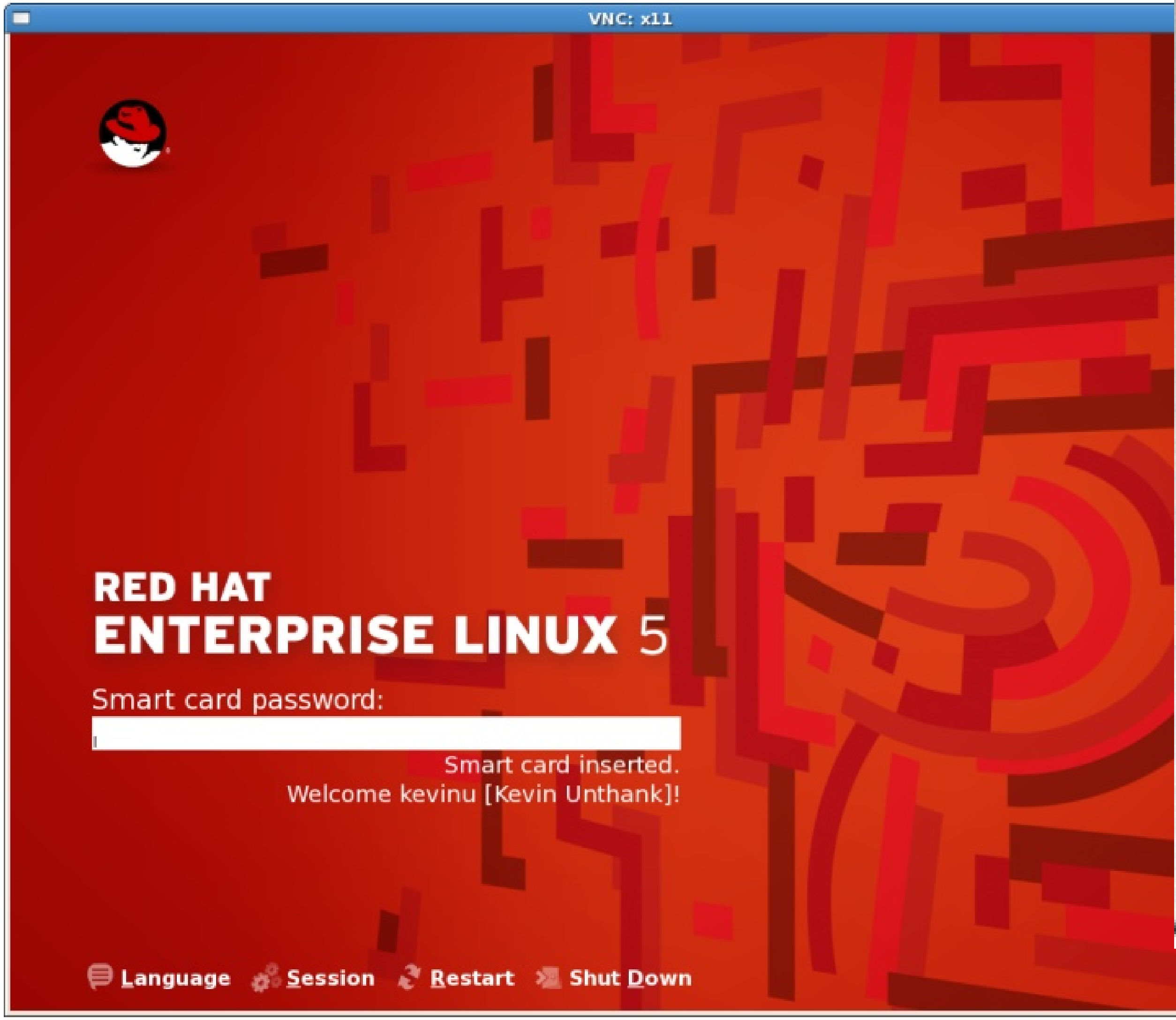
Time from a critical issue being known to the public until the day that a fix is available via RHN
Red Hat Enterprise Linux 4, Feb 2005-Feb 2006



Red Hat Security Certifications

- **NIAP/Common Criteria: The most evaluated operating system platform**
  - Red Hat Enterprise Linux 2.1 – EAL 2 (Completed: February 2004)
  - Red Hat Enterprise Linux 3 EAL 3+/CAPP (Completed: August 2004)
  - Red Hat Enterprise Linux 4 EAL 4+/CAPP (Completed: February 2006)
  - Red Hat Enterprise Linux 5 EAL4+/CAPP/LSPP/RBAC (Completed: June 2007)
- **DII-COE**
  - Red Hat Enterprise Linux 3 (Self-Certification Completed: October 2004)
  - Red Hat Enterprise Linux: First Linux platform certified by DISA
- **DCID 6/3**
  - Currently PL3/PL4: ask about kickstarts.
  - Often a component in PL5 systems
- **DISA SRRs / STIGs**
  - Ask about kickstarts.
- **FIPS 140-2**
  - Red Hat / NSS Cryptography Libraries certified Level 2

RHEL5 Security: Smart Card Support



RED HAT
ENTERPRISE LINUX 5

Smart card password:

Smart card inserted.
Welcome kevinu [Kevin Unthank]!

United States Government

JAN2009

Affiliation
Uniformed Services

Agency/Department
Army

Expires
2009JAN18

Doe,
John, G.

Pay Grade
E05

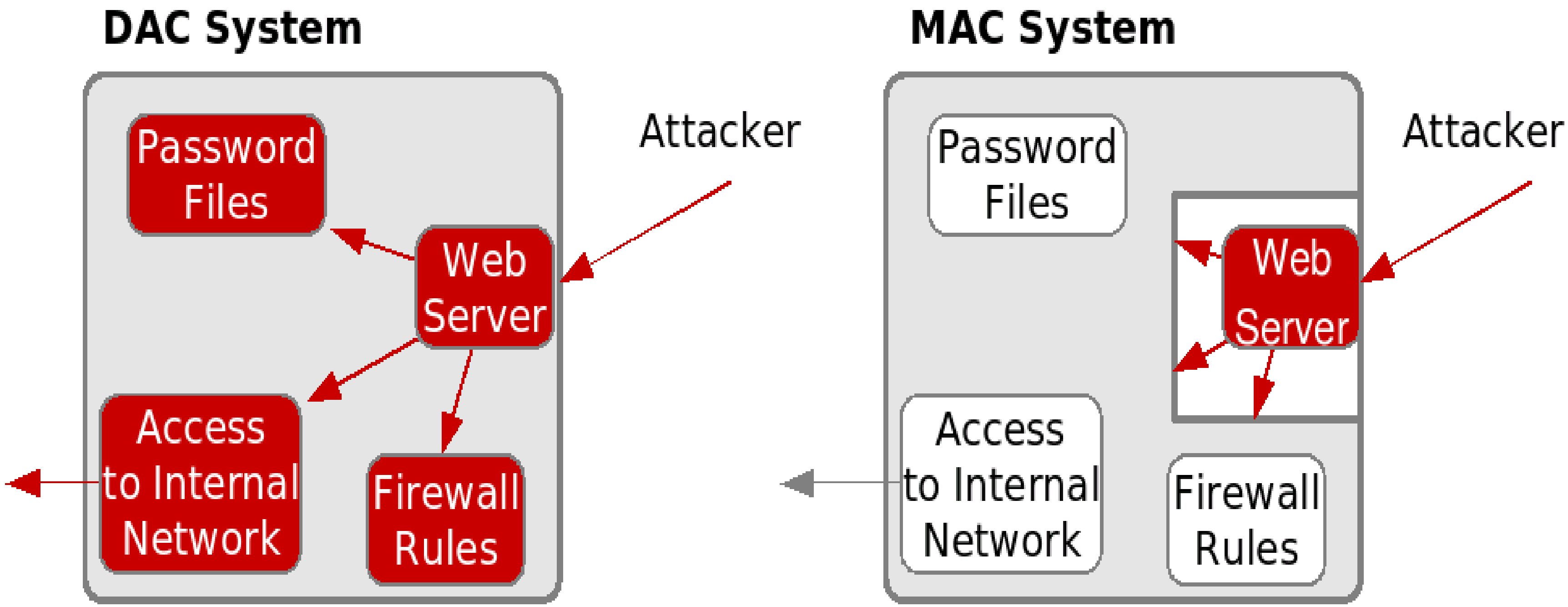
Rank
SGT

Geneva Conventions Identification Card

RHEL5 Security: NIST Standards Work

- Extensible Configuration Checklist Description Format (XCCDF)
  - Enumeration for configuration requirements
  - DISA FSO committed to deploying STIG as XCCDF
  - Others working with NIST
  - Security policy becomes one file

RHEL5 Security: Basics of SELinux



RHEL5 Security: SELinux Policies

- Targeted Policy (Default)
  - Applications run unconfined unless explicitly defined policy exists
- Strict Policy
  - All application actions explicitly allowed through SELinux, else actions denied
- MLS
  - Polyinstantiated file systems
  - Allows for different “views” based on clearance level

RHEL5 Security: MLS Requirements

- **Systems Must Be Tamperproof**

There must be no way for attackers or others on the system to intentionally or accidentally disable it or otherwise interfere with its operation

- **Systems Must Be Nonbypassable**

There must be no way to gain access to system resources except through mechanisms that use the reference monitor to make access control decisions

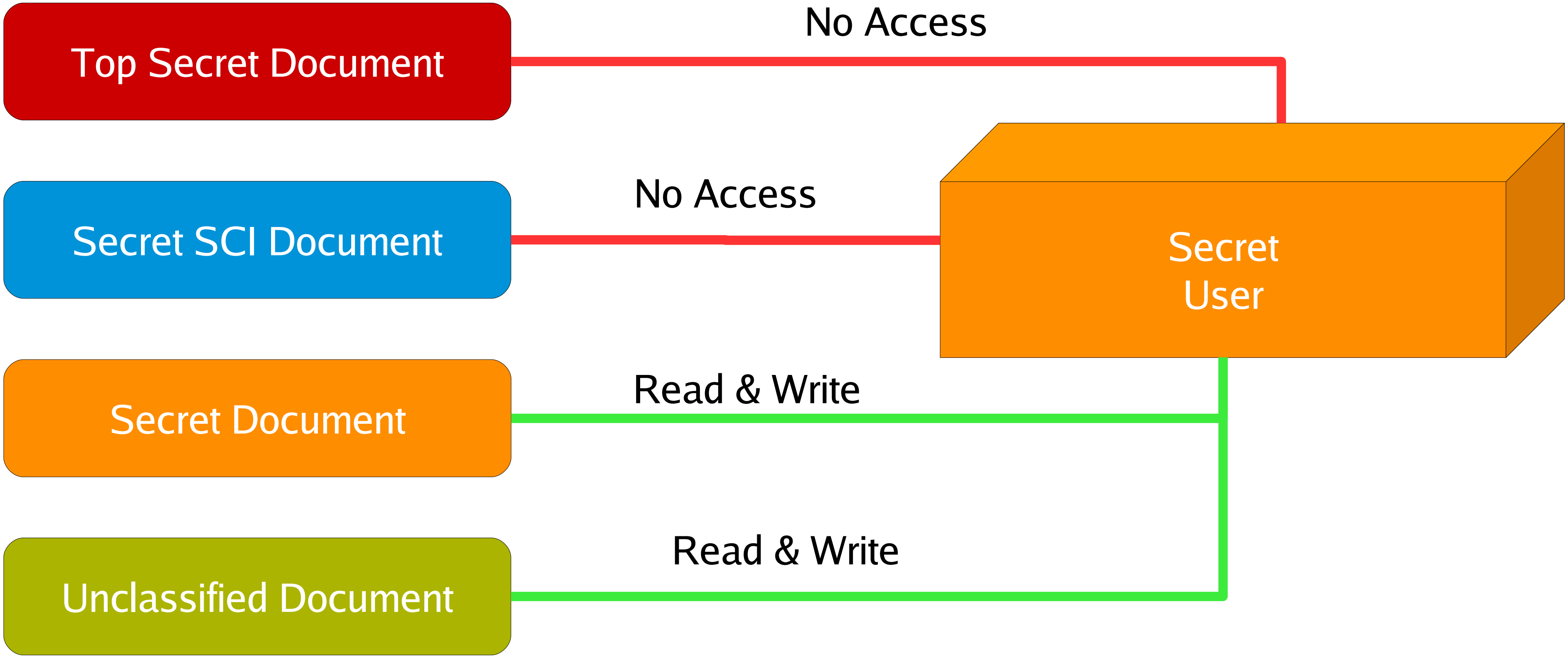
- **Access Must Be Verifiable**

There must be a way to convince third-party evaluators (i.e. Auditors) that the system will always enforce MLS correctly

- **No Covert Channels**

Eliminate footprints of other processes on the system (process threads, resource utilization, disk activities, etc)

RHEL5 Security: MLS Design



RHEL5 Security: Use of MLS

- **Multi-Level Servers**

Servers implemented which connects to networks of differing classification, utilizing MLS for network traffic monitoring

- **One-way Transfer (OWT)**

Devices with transfer from“low” to “high”

- **Downgrading Guards**

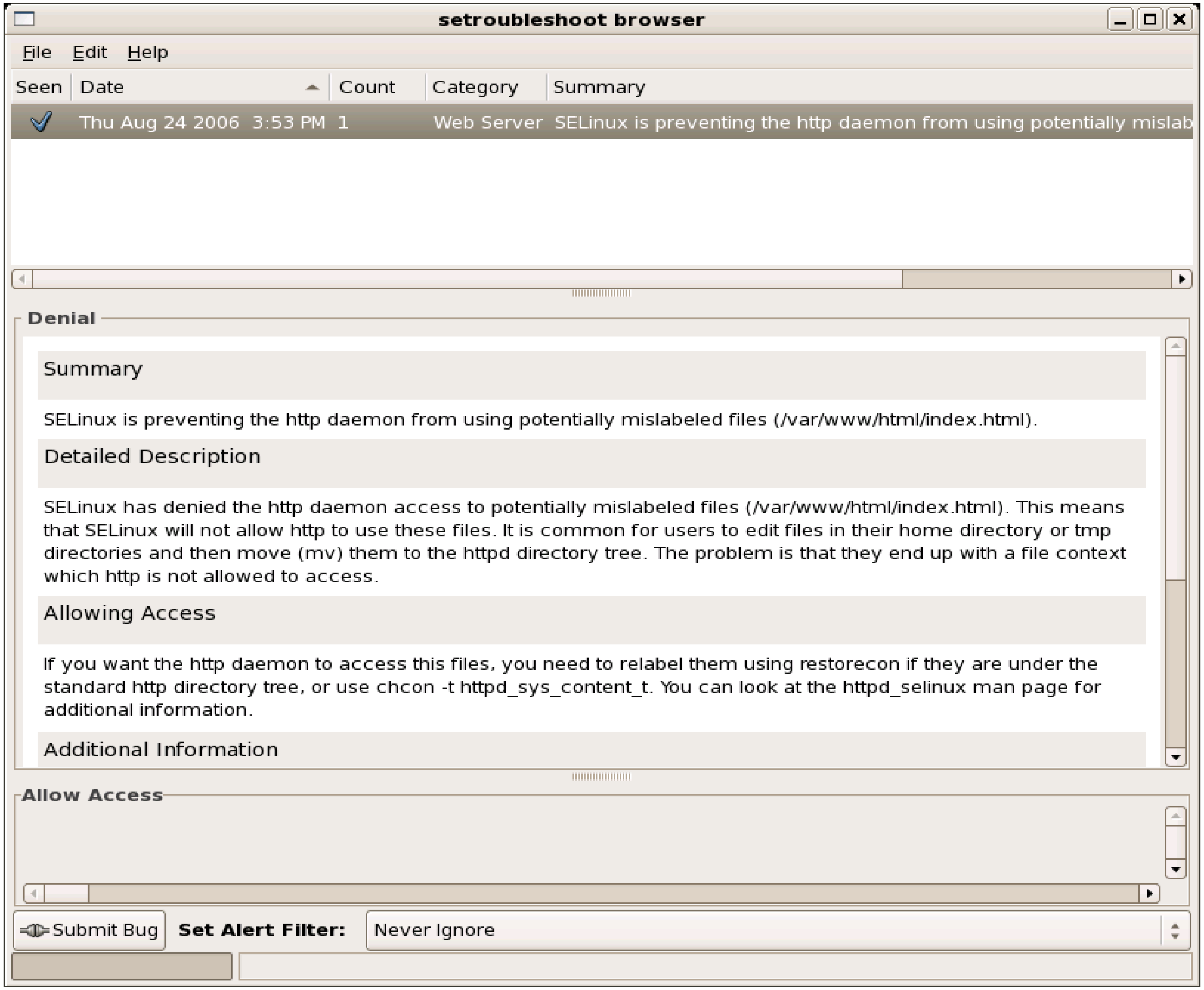
Devices which transfer data in any direction between networks of differing classifications

RHEL5 SELinux Enhancements

- **Expanded SELinux targeted policy coverage**
  - Provides coverage for all core system services, versus 11 in Red Hat Enterprise Linux 4
  - Includes support for Multi Level Security (MLS) enforcement model
    - In addition to existing RBAC and TE models
- **An additional level of protection against security exploits**
  - Fine-grained policies via kernel-enforced mandatory access controls
  - Limits the scope of security vulnerabilities
  - Beyond what any other general-purpose OS can deliver

RHEL5 SELinux Enhancements

- New troubleshooting tool provides clear, easy-to-understand, GUI-based, security violation notifications
- Over 60 events defined today



Seen	Date	Count	Category	Summary
✓	Thu Aug 24 2006 3:53 PM	1	Web Server	SELinux is preventing the http daemon from using potentially mislab

Denial

Summary

SELinux is preventing the http daemon from using potentially mislabeled files (/var/www/html/index.html).

Detailed Description

SELinux has denied the http daemon access to potentially mislabeled files (/var/www/html/index.html). This means that SELinux will not allow http to use these files. It is common for users to edit files in their home directory or tmp directories and then move (mv) them to the httpd directory tree. The problem is that they end up with a file context which http is not allowed to access.

Allowing Access

If you want the http daemon to access this files, you need to relabel them using restorecon if they are under the standard http directory tree, or use chcon -t httpd_sys_content_t. You can look at the httpd_selinux man page for additional information.

Additional Information

Allow Access

Set Alert Filter: Never Ignore

RHEL5 SELinux Enhancements

- Greatly improved logging, with easy-to-decipher information

OLD: Red Hat Enterprise Linux 4 /var/log/messages entry

```
time->Thu Aug 24 15:50:58 2006
type=AVC_PATH msg=audit(1156449058.917:552):
path="/var/www/html/index.html"
type=SYSCALL msg=audit(1156449058.917:552): arch=40000003 syscall=196
success=no exi
t=-13 a0=8d4d4d0 a1=bfb5e97c a2=434ff4 a3=2008171 items=0 ppid=23799
pid=23805 auid=3267 uid=48 gid=48 euid=48 suid=48 fsuid=48 egid=48 sgid=48 fsgid=48
tty=(none) comm="httpd" exe="/usr/sbin/httpd" subj=user_u:system_r:httpd_t:s0 key=(null)
type=AVC msg=audit(1156449058.917:552): avc: denied { getattr } for pid=23805 com
m="httpd" name="index.html" dev=dm-0 ino=6260297
scontext=user_u:system_r:httpd_t:s0
 tcontext=system_u:object_r:user_home_t:s0 tclass=file
```

NEW: Red Hat Enterprise Linux 5 /var/log/messages entry

```
Aug 24 15:53:10 localhost /usr/sbin/setroubleshootd:     SELinux is
preventing /usr/sbin/httpd "getattr" access to /var/www/html/index.html.
See audit.log for complete SELinux messages.
```

RHEL5 SELinux Enhancements

- **ExecShield**
  This enhancement can prevent any memory that was writable from becoming executable. This prevents an attacker from writing his code into memory and then executing it

- **Stack Smashing protection (Canary values)**
  The system will place a canary value at a randomized point above the stack. This canary value is verified during normal operation. If the stack has been smashed, the canary value will have been overwritten, indicating that the stack has been smashed. This is a method to detect buffer overflows early.

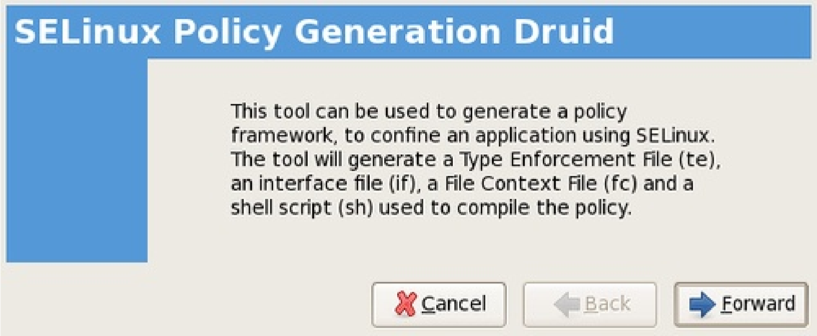
- **FORTIFY_SOURCE GCC option**
  When the compiler knows the size of a buffer, functions operate on the buffer to make sure it will not overflow at runtime. This works to help catch format string flaws as well as buffer overflows.

RHEL5 SELinux Enhancements

Policy creation now a two-step process

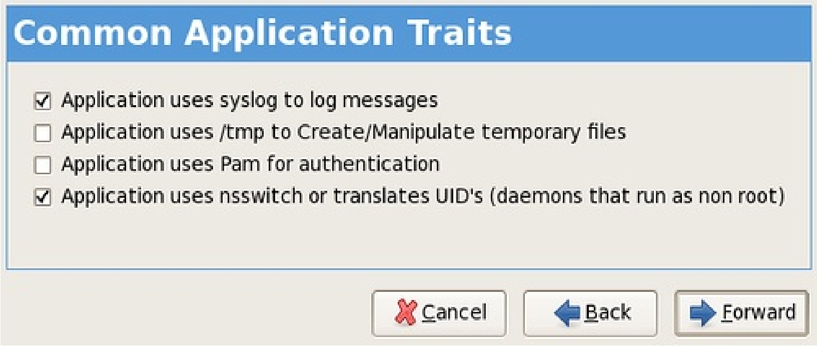
1) system-config-selinux
   - Creates template policy (network, filesystem read/write, etc)

2) audit2allow
   - Traces application, ensuring proper accesses



SELinux Policy Generation Druid

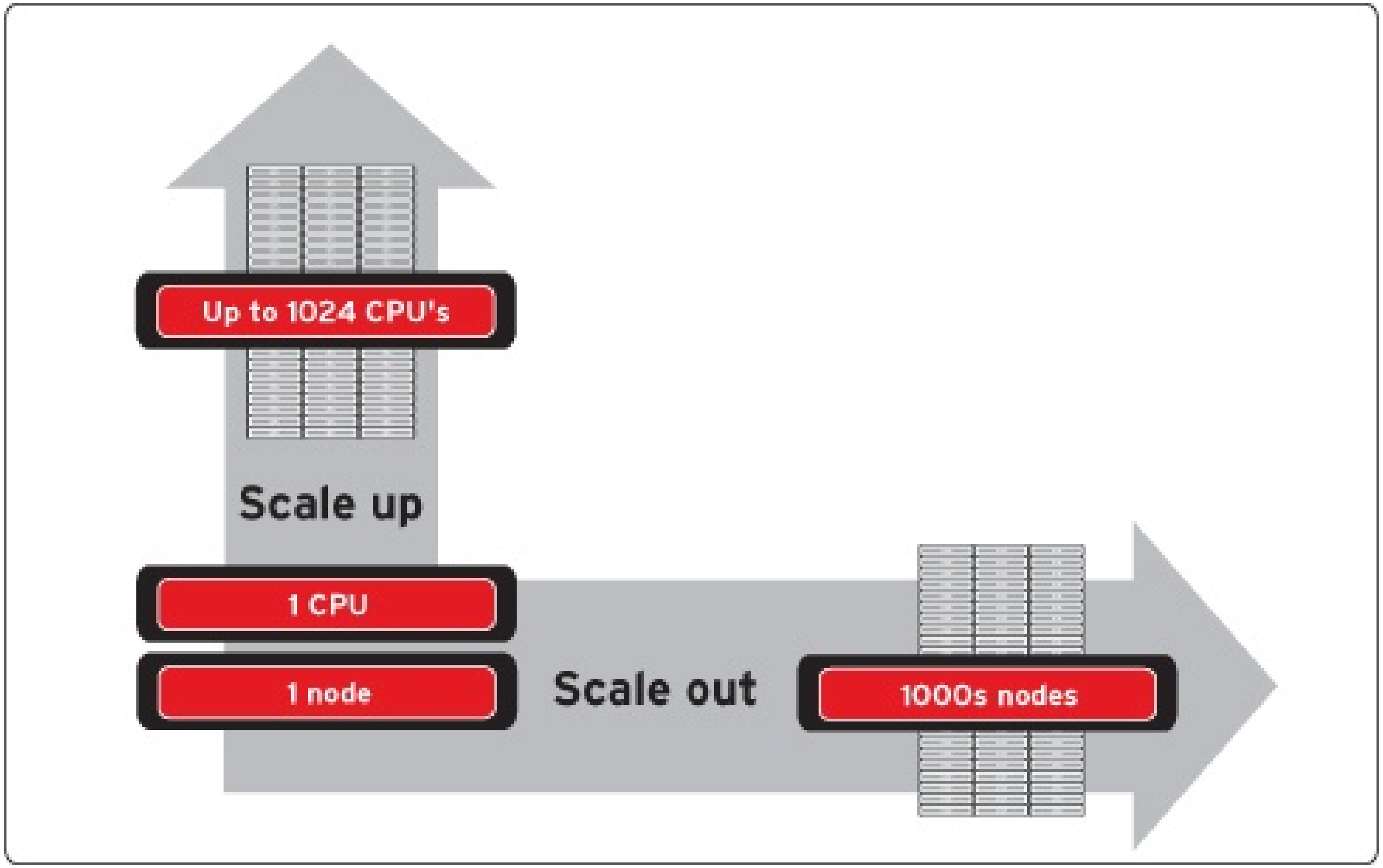
This tool can be used to generate a policy framework, to confine an application using SELinux. The tool will generate a Type Enforcement File (te), an interface file (if), a File Context File (fc) and a shell script (sh) used to compile the policy.



Common Application Traits

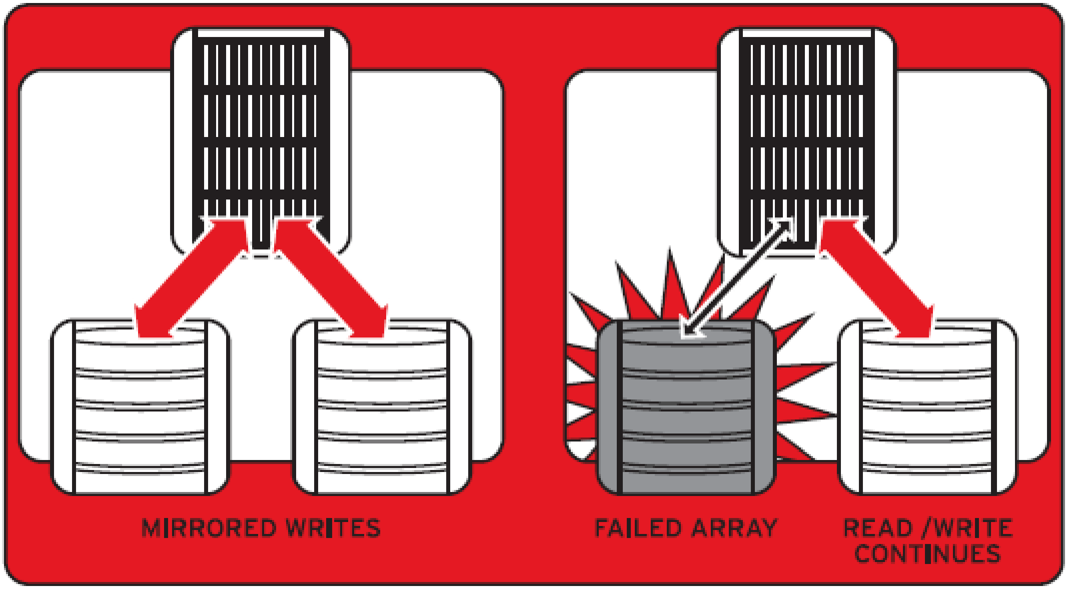
☑ Application uses syslog to log messages
□ Application uses /tmp to Create/Manipulate temporary files
□ Application uses Pam for authentication
☑ Application uses nsswitch or translates UID's (daemons that run as non root)

RHEL5 High Availability: Mature Scaling



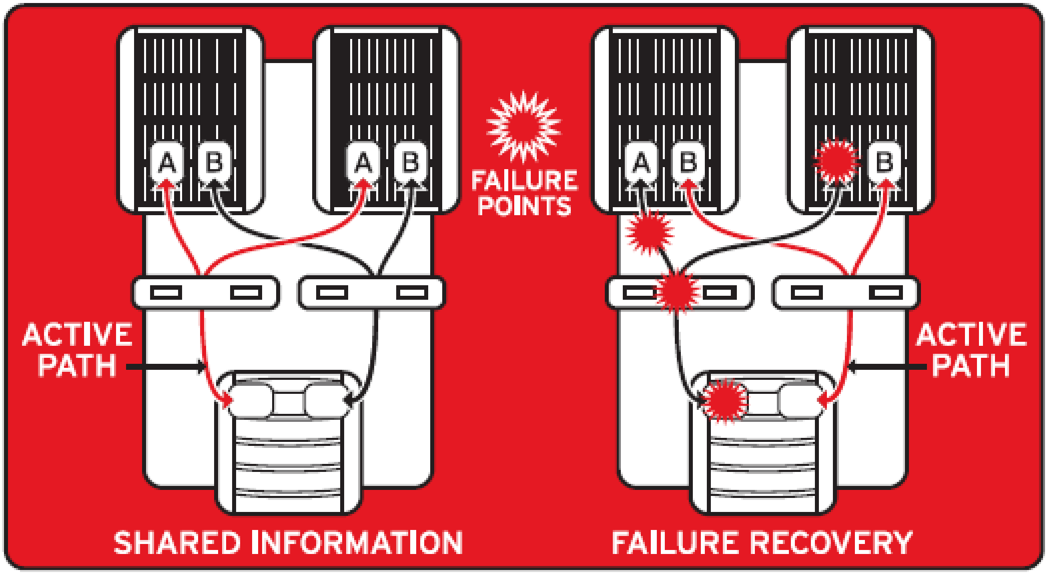
LVM Host-Based Synchronous Mirroring

- Each write is simultaneously written to 2 or more local or SAN disks (RAID1)
- LVM automatically detects failure, uses the identical, mirrored disks or LUN
- Upon restoration, recovery process begins in background
- If minor outage, transaction log rapidly replays missed I/O



Device Mapper Multipath IO (MPIO)

- Connects & manages multiple paths through SAN to storage array
- Upon component failure, MPIO redirects traffic via redundant pathing
- Active/Active array support
- Bundled into RHEL

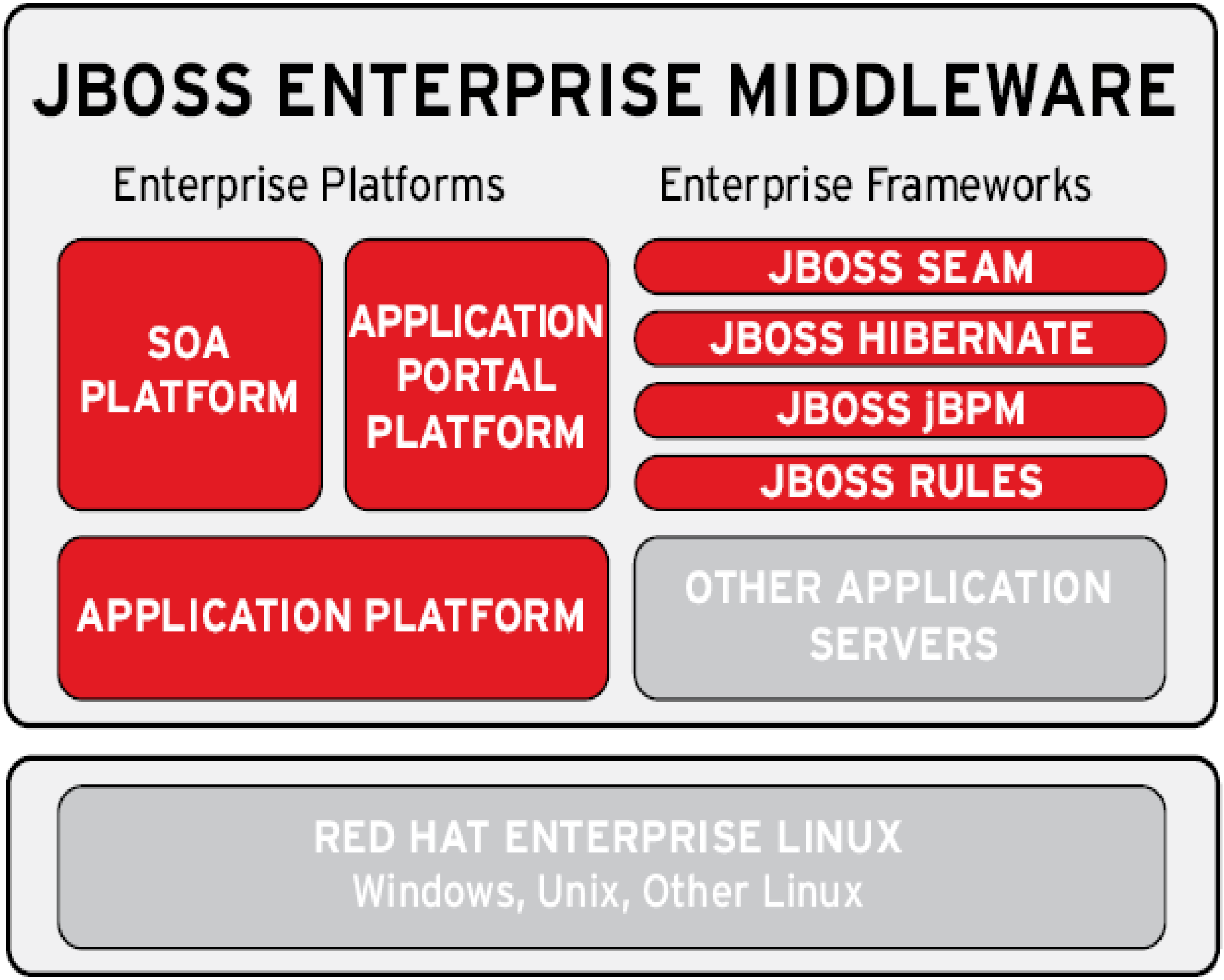


GFS

- Red Hat GFS is an open source, POSIX-compliant, cluster file system.
- It provides a consistent file system image across the server nodes in a cluster, allowing Red Hat Enterprise Linux servers to simultaneously read and write to a single shared filesystem.
- Red Hat GFS also includes Red Hat Cluster Suite providing integrated application high availability and failover.

Red Hat SOA Direction

- JBoss
- MetaMatrix



JBOSS ENTERPRISE MIDDLEWARE

Enterprise Platforms

Enterprise Frameworks

SOA PLATFORM

APPLICATION PORTAL PLATFORM

JBOSS SEAM

JBOSS HIBERNATE

JBOSS jBPM

JBOSS RULES

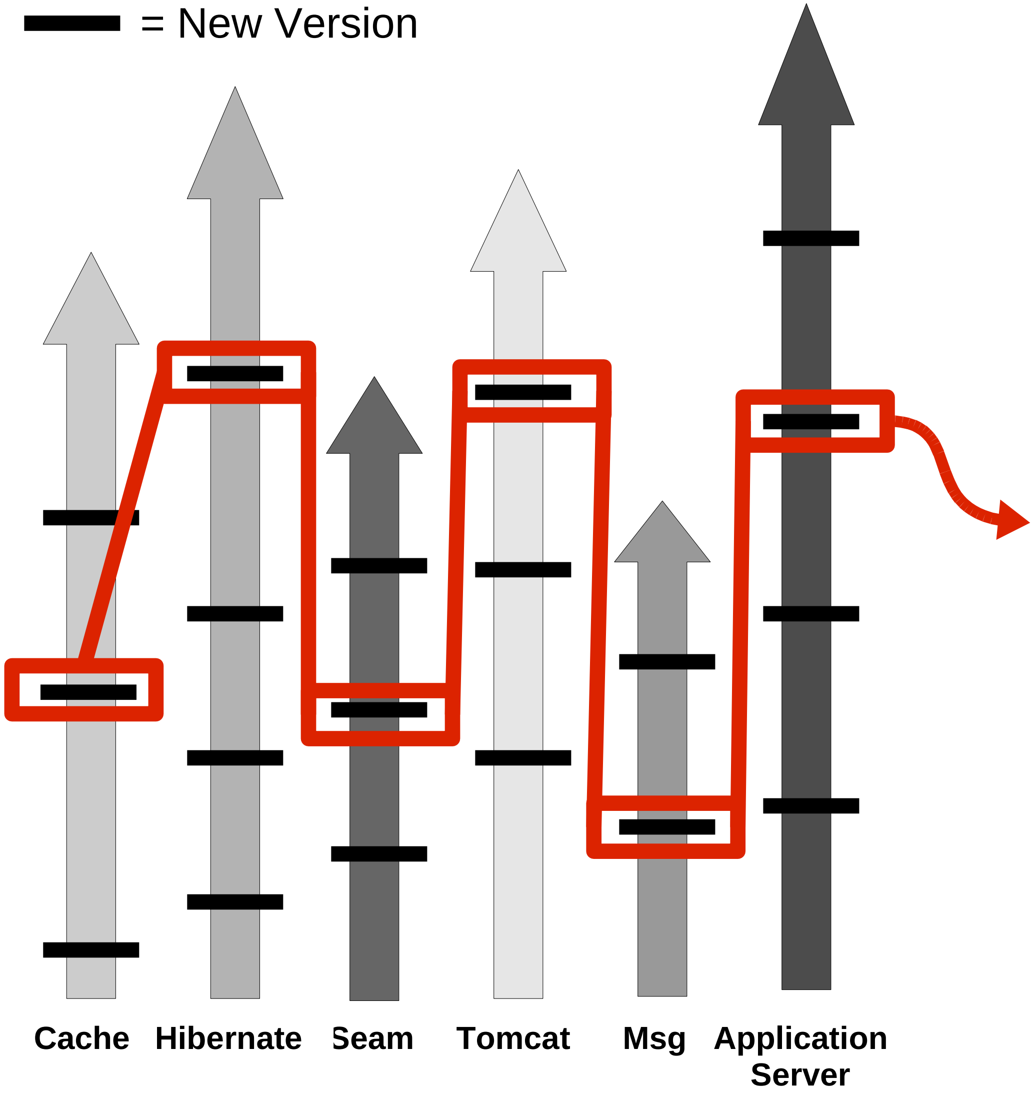
APPLICATION PLATFORM

OTHER APPLICATION SERVERS

RED HAT ENTERPRISE LINUX

Windows, Unix, Other Linux

JBoss Enterprise: Stability & Performance



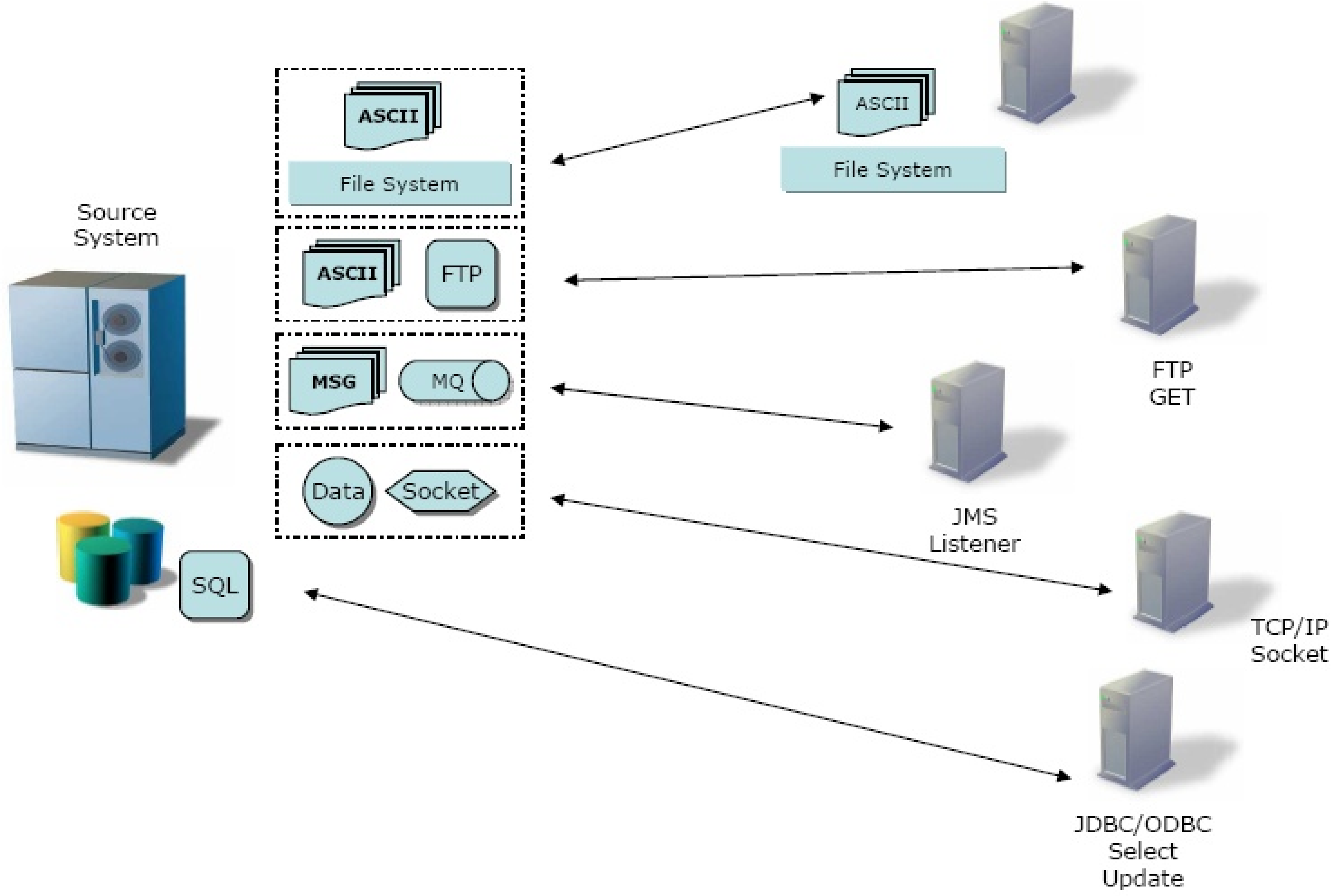
Corporate Challenge:

- Integrate Multiple Versions
- Coordinate 100,000 developers

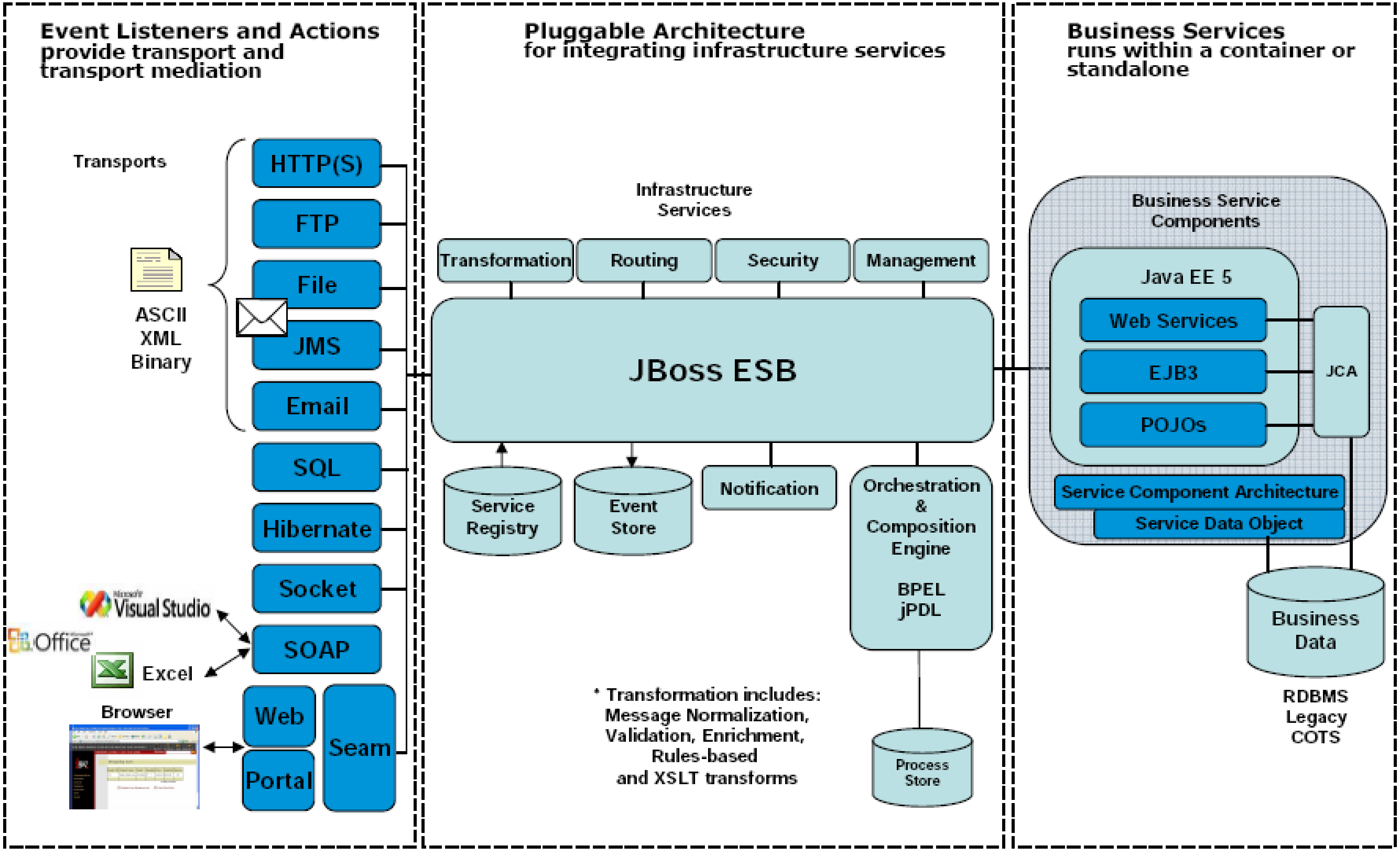
Solution: JBoss Enterprise Platforms

- Single, integrated, certified distributions
- Extensive Q/A Process
- Industry-leading Support
- Documentation
- Secure, Production-level Configurations
- Multi-year Errata Policy

JBoss ESB: Before (Stovepipes)

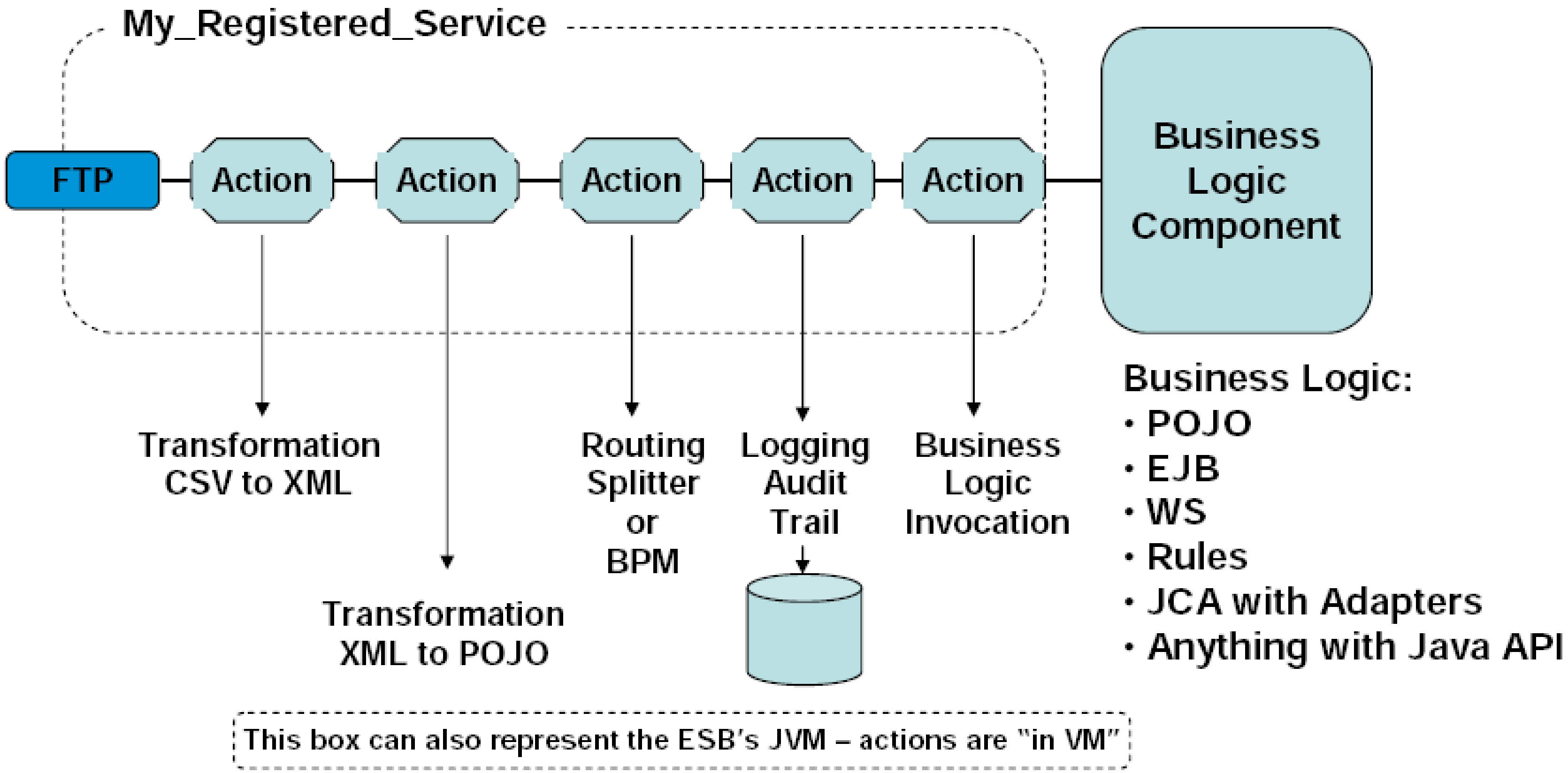


JBoss ESB: After (Anything-To-Anything)

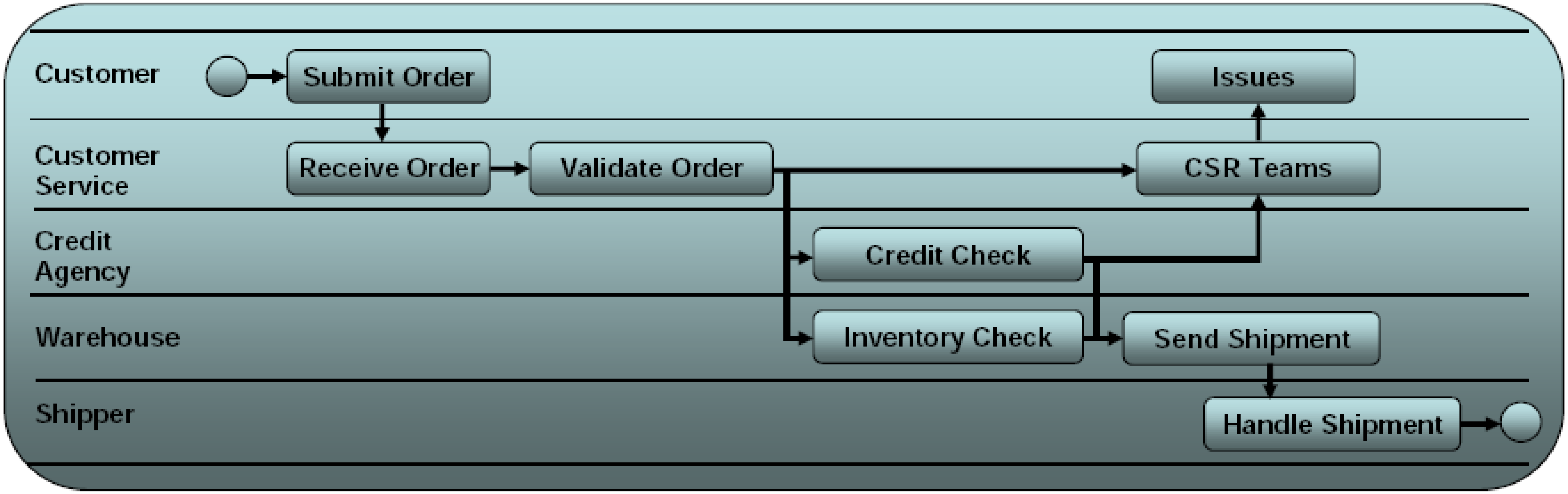

Event Listeners and Actions provide transport and transport mediation
Transports
ASCII
XML
Binary
HTTP(S)
FTP
File
JMS
Email
SQL
Hibernate
Socket
Visual Studio
Office
Excel
SOAP
Browser
Web
Portal
Seam
Pluggable Architecture for integrating infrastructure services
Infrastructure Services
Transformation
Routing
Security
Management
JBoss ESB
Service Registry
Event Store
Notification
Orchestration & Composition Engine
BPEL
jPDL
Process Store
* Transformation includes: Message Normalization, Validation, Enrichment, Rules-based and XSLT transforms
Business Services runs within a container or standalone
Business Service Components
Java EE 5
Web Services
EJB3
POJOs
JCA
Service Component Architecture
Service Data Object
Business Data
RDBMS
Legacy
COTS

JBoss Action Pipeline

- Actions are reusable components that can be chained together to form capabilities of a registered service. Can be dynamically added/removed at runtime.



JBoss Rules



Customer

Customer Service

Credit Agency

Warehouse

Shipper

Submit Order

Receive Order

Validate Order

CSR Teams

Issues

Credit Check

Inventory Check

Send Shipment

Handle Shipment

Validate Order

a Parse XML
b Transform
c Apply Business Rules

Credit Check

a Create Outbound Msg
b Handle Response
c Apply Business Rules

Send Shipment

a Determine Shipper(s)
b Print Labels
c Print Pick Tickets
d Create & Send ASNs

Inventory Check

a Send to N Warehouses
b Handle N Responses
c Determine Best WHSEs
d Handle Drop-Ships

ESB Mediates & Provides Services

JBoss Conclusion

- Open Source makes JBoss SOA happen
  - ESB
  - Hibernate
  - SEAM
- ESB is a solution for application/process integration
- ESB is about service intermediation
- Ask about getting evaluation copies!

MetaMatrix Background

- On-demand access to distributed information
  - Real time integration of diverse data
  - Avoid unnecessary data replication

- Metadata-Driven
  - Integration in days, not weeks
  - Reduce “long tail” of application maintenance

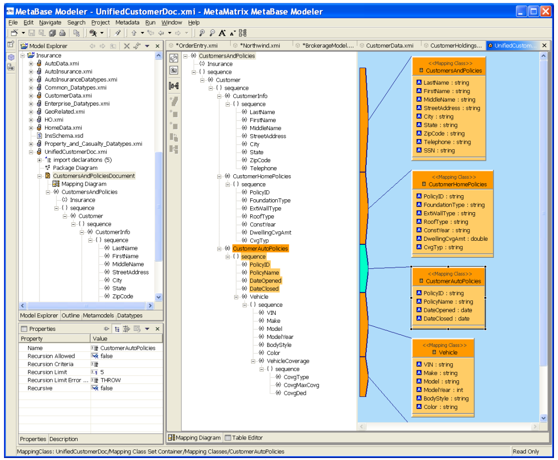
- Improved data agility

- Abstracts data into a “single view” without need to move data between databases
  - Single view of Customer – CRM
  - Single view of Supplier – Supply Chain
  - Single view of Employee – HR Consolidation

- SOA Enabler
  - Consume/Produce Web Services
  - .... and still provide support for ODBC, JDBC, and legacy!

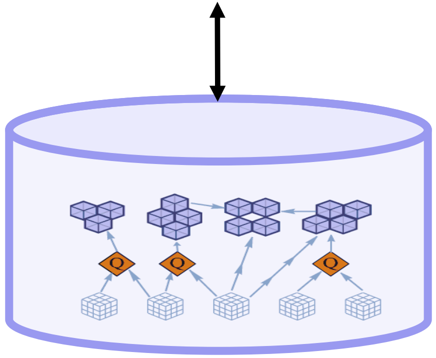
MetaMatrix: How it works

MetaMatrix Designer

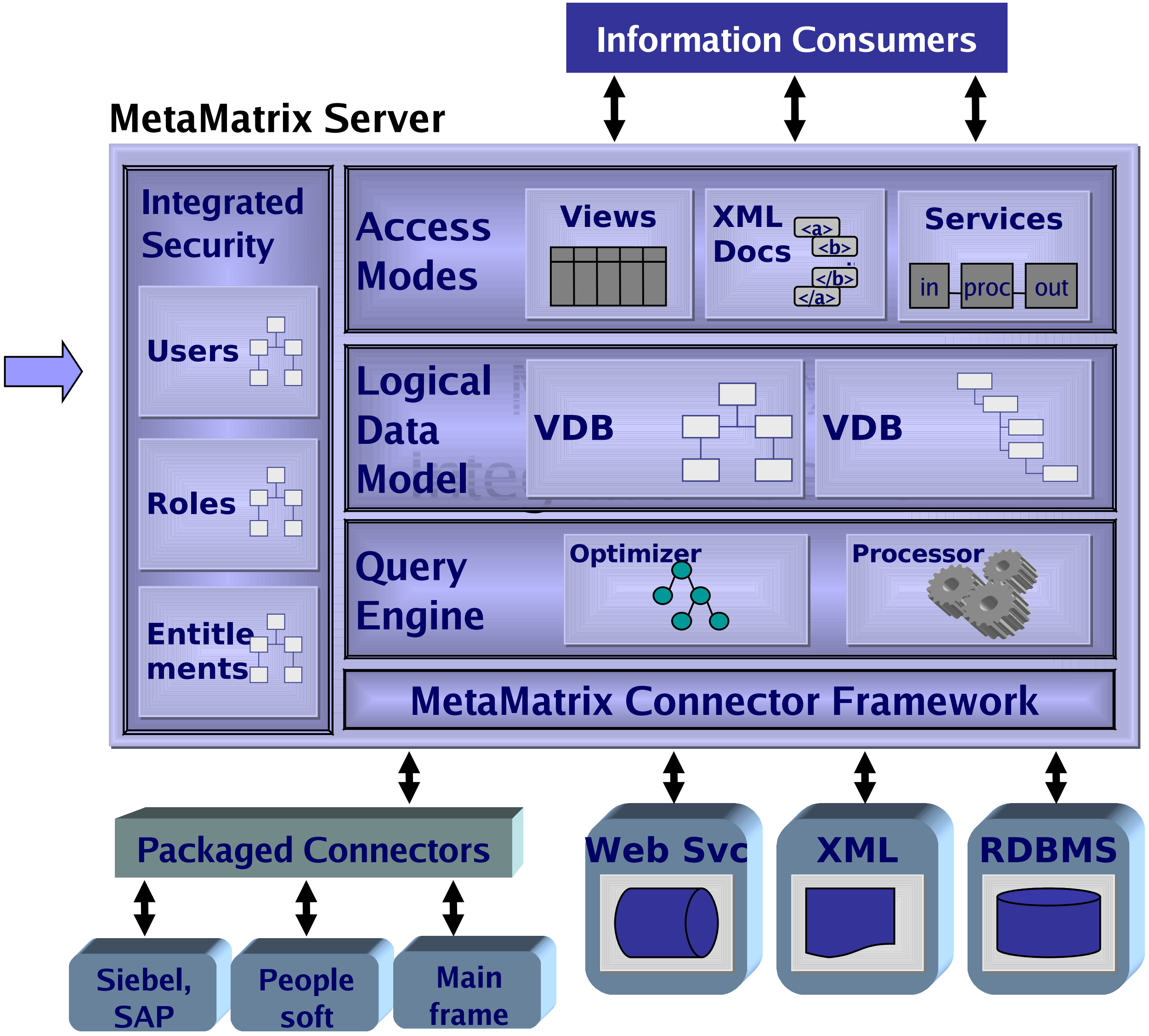


Property	Value
Name	CustomerAutoPolicies
Recursion Allowed	false
Recursion Criteria	
Recursion Limit	5
Recursion Limit Error ...	THROW
Recursive	false

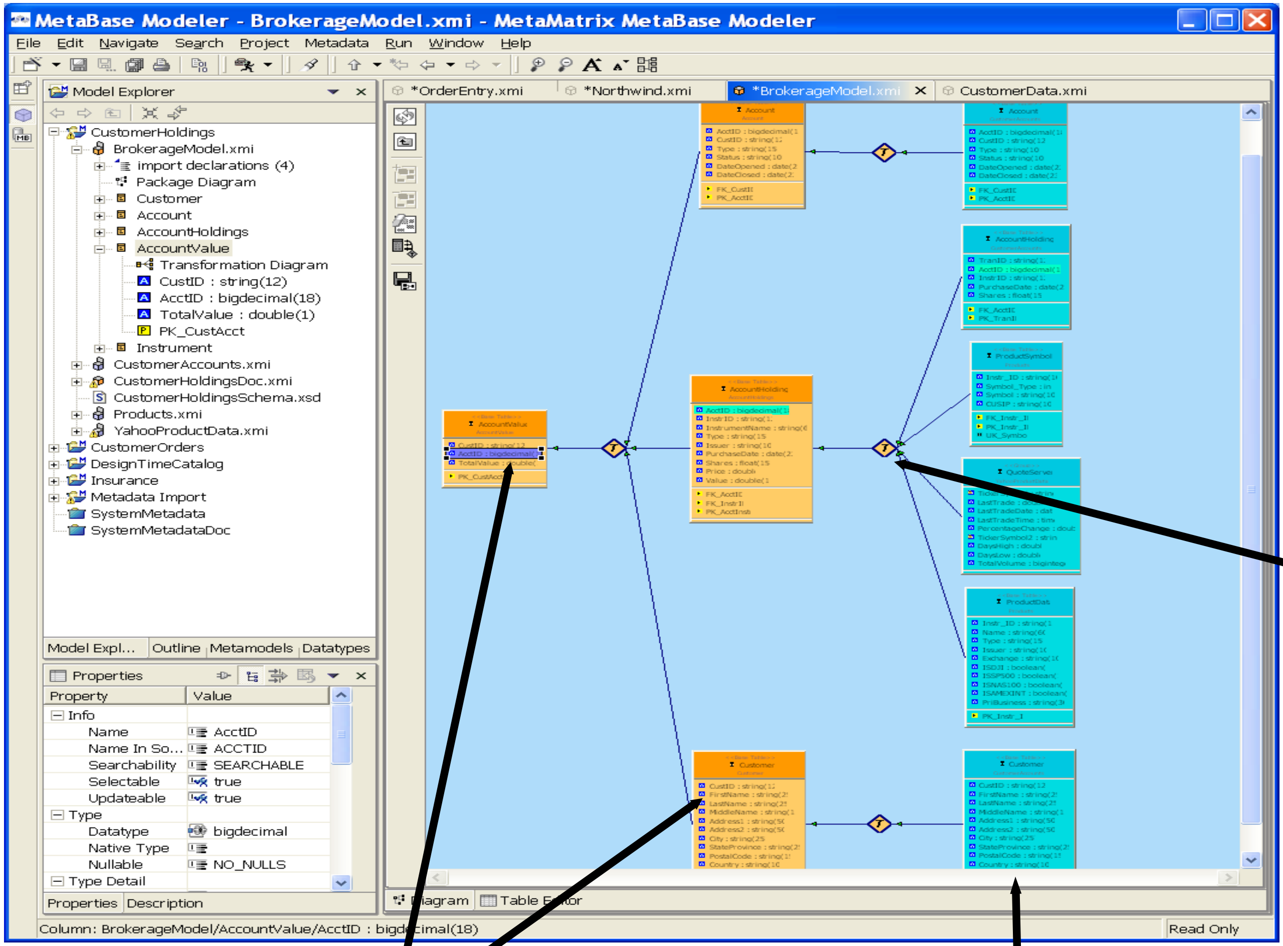
MappingClass: UnifiedCustomerDoc/Mapping Class Set Container/Mapping Classes/CustomerAutoPolicies



Metadata Repository



MetaMatrix Enterprise Designer



- CustomerHoldings
  - BrokerageModel.xmi
    - import declarations (4)
    - Package Diagram
    - Customer
    - Account
    - AccountHoldings
    - AccountValue
      - Transformation Diagram
      - CustID : string(12)
      - AcctID : bigdecimal(18)
      - TotalValue : double(1)
      - PK_CustAcct
    - Instrument
  - CustomerAccounts.xmi
  - CustomerHoldingsDoc.xmi
  - CustomerHoldingsSchema.xsd
  - Products.xmi
  - YahooProductData.xmi
- CustomerOrders
- DesignTimeCatalog
- Insurance
- Metadata Import
- SystemMetadata
- SystemMetadataDoc

Property	Value
Info	
Name	AcctID
Name In So...	ACCTID
Searchability	SEARCHABLE
Selectable	true
Updateable	true
Type	
Datatype	bigdecimal
Native Type	
Nullable	NO_NULLS
Type Detail	

Column: BrokerageModel/AccountValue/AcctID : bigdecimal(18)   Read Only

Defined by Models, not code

Transformations:

- Select
- Join/Aggregate
- Filter
- Functions
- Text/String
- Numeric
- Decode
- User Defined

Logical Models

Physical Models representing actual data sources

Red Hat Identity Management

- Directory Server
- Certificate Server

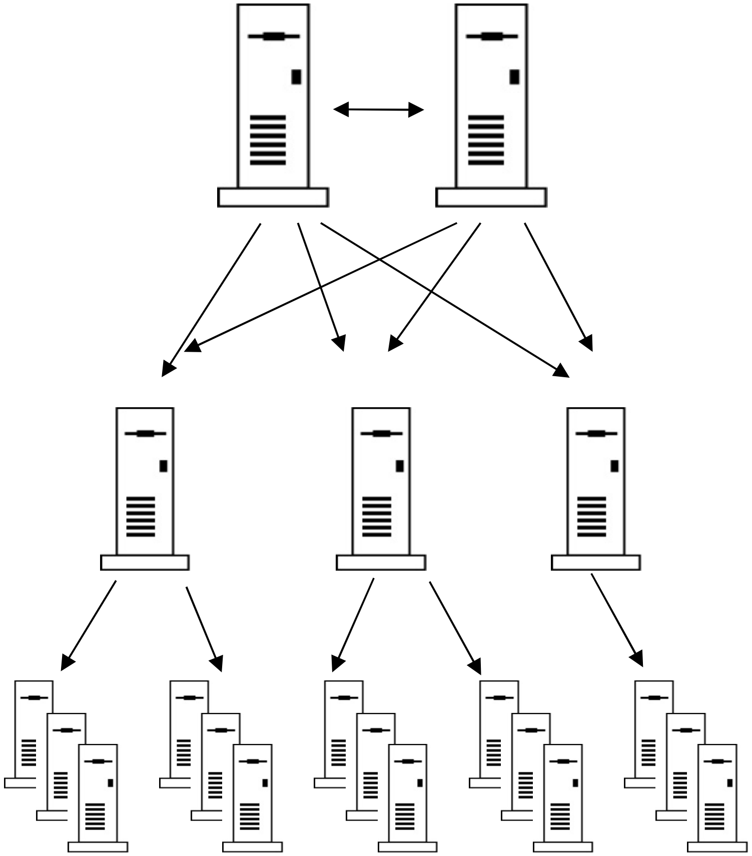
Red Hat Authentication History

- On December 8, 2004, Red Hat acquired assets from AOL's Netscape Security Solutions business unit, including currently shipping products:
  - Netscape Certificate Management System (Red Hat Certificate System)
  - Netscape Directory Server (Red Hat Directory Server)

- Acquisition of JBoss in June 2006 now provides an extension for the identity management technologies into the Application and Web Services space.

What does Red Hat Directory Server provide?

- Standards Compliance (LDAP v2/v3)
- High Performance
  - Multi-Master Replication
  - WAN Replication
  - Load Balancing
  - Data Redundancy
  - Fault Tolerance
- Windows User Synchronization
- SASL Authentication
- Fine Grained ACL



What does Red Hat Certificate System provide?

- Standards Based PKI
- Unmatched Scalability
  - 14,000 certificate issuances/hour
  - 12M certificates issued < 35 days
  - Largest public CRL: 1.2M
- Government Support
  - FIPS-140 Certified
  - EAL4/CIMC Protection Profile
- Replaces Passwords, Single Sign-On
- Smart Card Support
- Integrated with RHEL & Windows

Certificate Server Scalability

- 25M+ Certificates issued from the DoD PKI CAs, 1999-Present
- 3.5M – 4M “active” smart cards, most with 3 certificates
- Issuing lifetime of individual CAs governed by the size of the database managed by the CA
  - As number of certificates issued increased, number of revocations increased, resulting in large CRLs
  - CMS 4.2 and CMS 6.1 on Solaris – approximately 1.5M issued certificates per CA
  - Red Hat CS 7.1 on Linux [deployed in 2006] – DT&E Testing prior to fielding demonstrated 3M issued certificates per CA with large CRL

PKI and Kerberos: Independently Successful

- PKI
  - Smart card authentication
  - Web services
    - TLS, SSL
  - Encryption
  - Signing
  - Data integrity
  - Non Repudiation
  - Asymmetric keys

- Kerberos
  - System Login
  - Secure Filesystem access
    - NFSv4, CIFS
  - Email server access
  - Printing
  - Symmetric keys

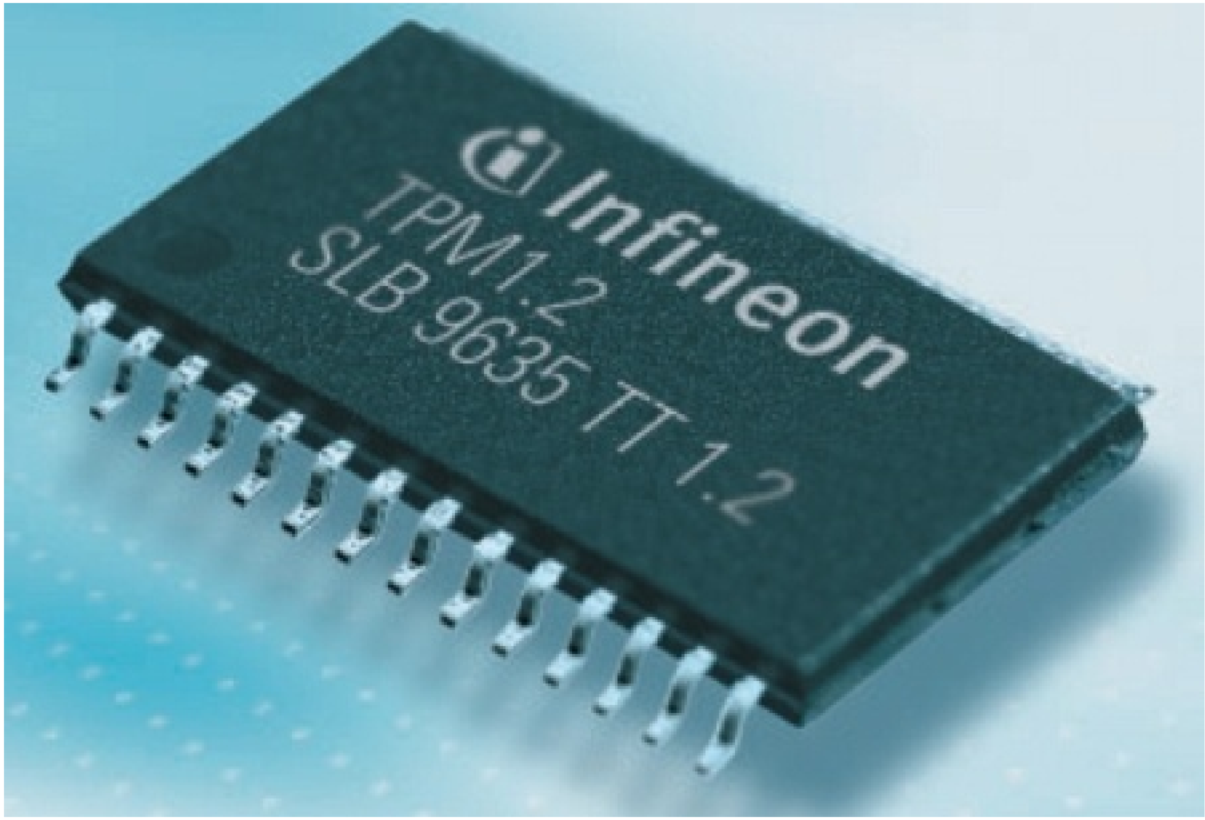
Directory & Certificate Server tie them together

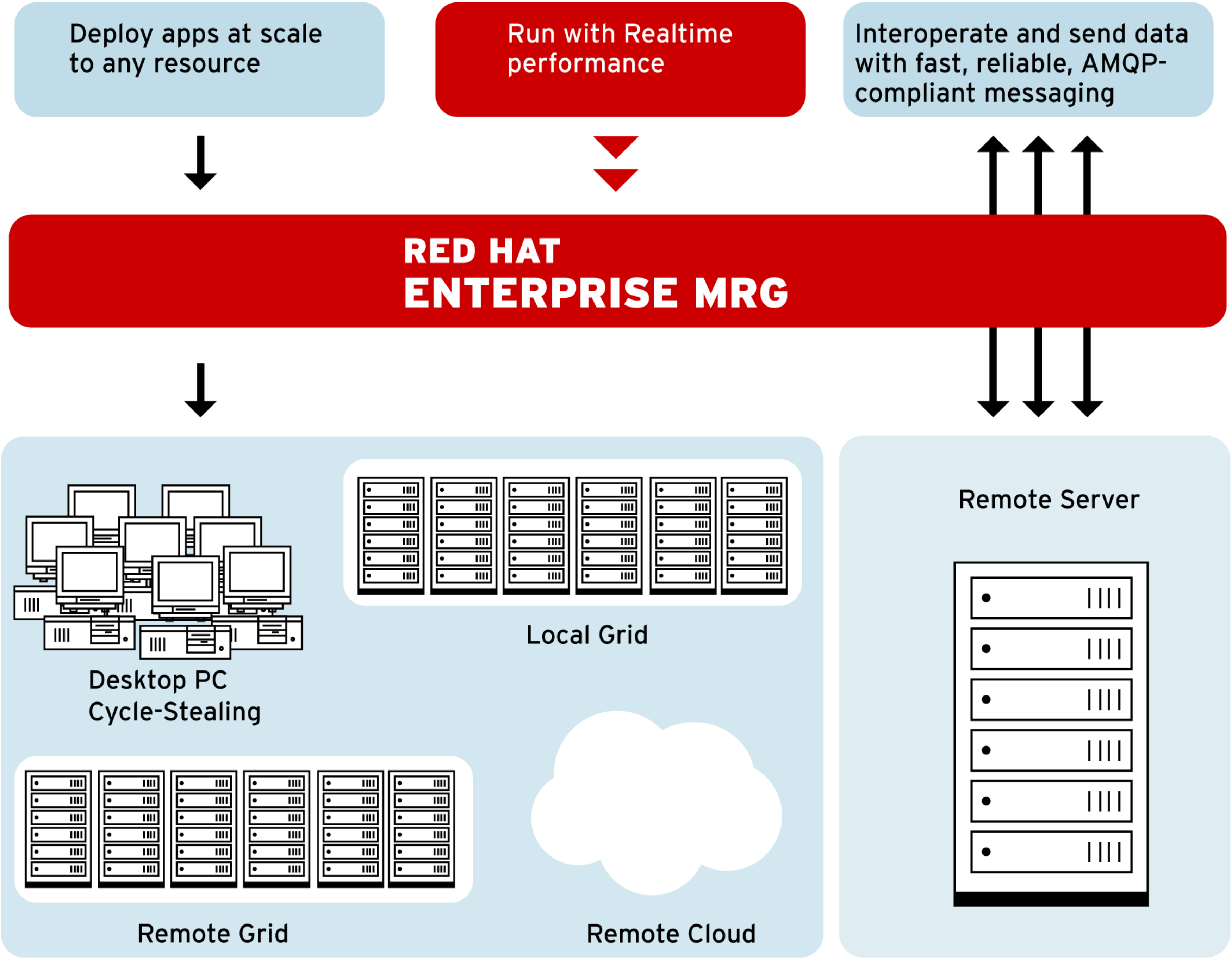
Red Hat Emerging Technologies

- Virtualization Updates
- MRG/Realtime
- freeIPA

Trusted Platform Module

- Planned for inclusion in next minor release of RHEL (5.2)
  - What functionality would you like to see?
  - What functionality would you use?





Deploy apps at scale to any resource

Run with Realtime performance

Interoperate and send data with fast, reliable, AMQP-compliant messaging

RED HAT
ENTERPRISE MRG

Desktop PC Cycle-Stealing

Local Grid

Remote Grid

Remote Cloud

Remote Server

MRG Realtime

- **Determinism**
  Ability to schedule high priority tasks predictably and consistently
- **Priority**
  Ensure that highest priority applications are not blocked by low priority
- **Quality Of Service** (QoS)
  Trustworthy, consistent response times
- **Proven results**
  - Average of 38% improvement over stock RHEL5
  - Timer event precision enhanced to µs level, rather than ms

MRG: Messaging

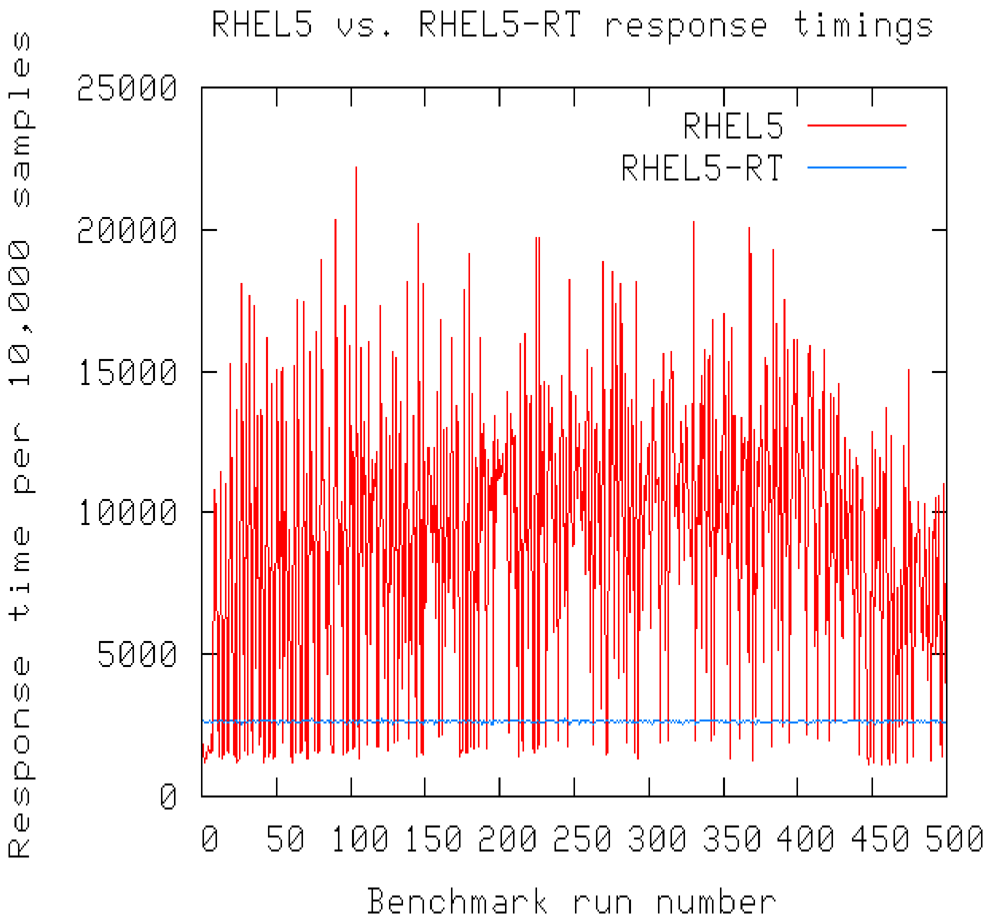
- Provides messaging that is up to 100-fold faster than before
- Spans fast messaging, reliable messaging, large-file messaging
- Implements AMQP, the industry's first open messaging standard, for unprecedented interoperability that is cross-language, cross-platform, multi-vendor, spans hardware and software, and extends down to the wire level
- Uses Linux-specific optimizations to achieve optimal performance on Red Hat Enterprise Linux and MRG Realtime
  - Takes advantage of RHEL clustering, IO, kernel, and more
  - Includes new high-performance AIO Journal for durable messaging
  - Provides native infiniband support for transient messaging

About AMQP

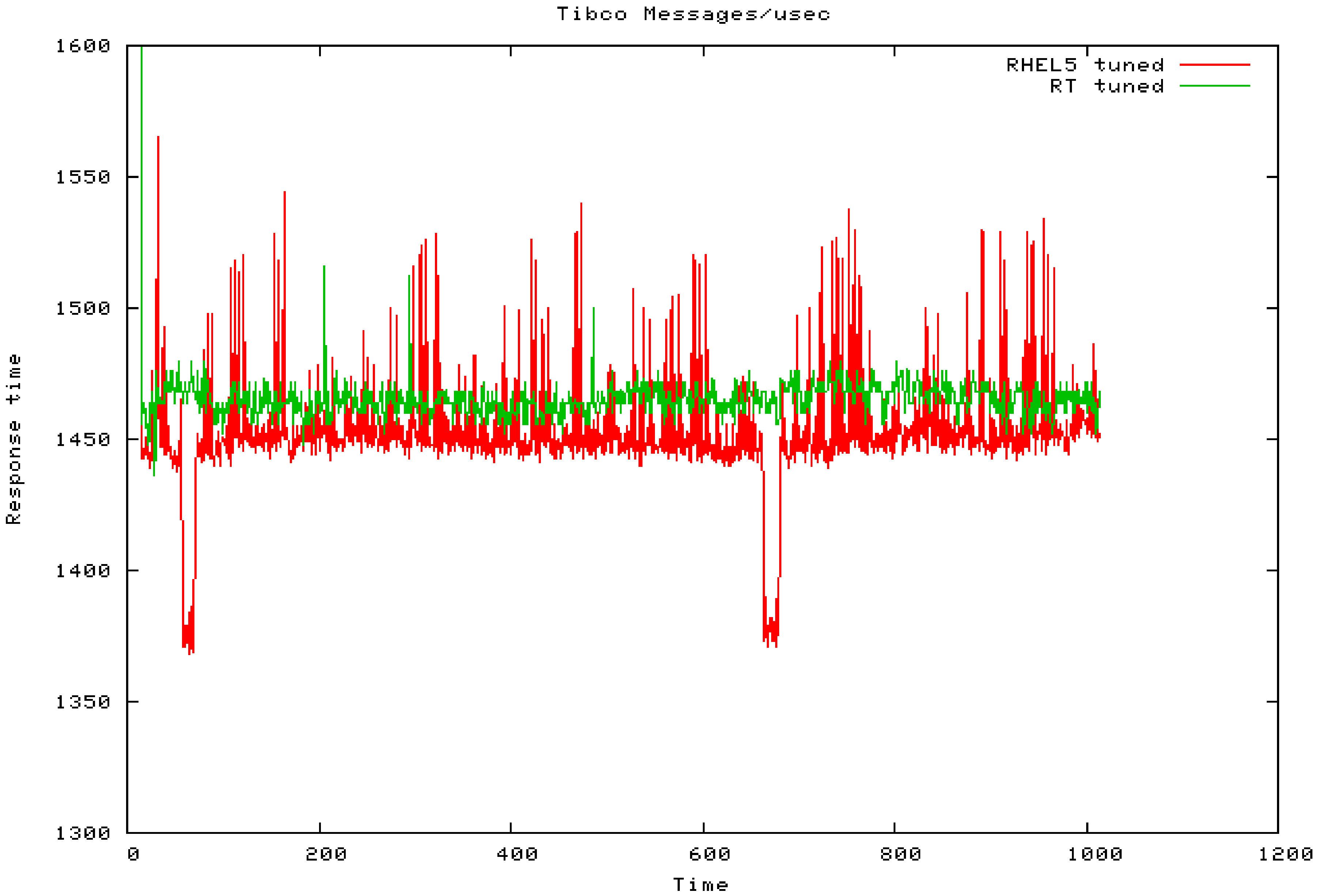
- AMQP is an open specification for messaging
  - It is a complete specification
  - Anyone may use the AMQP specification to create useful implementations without being charged for the IP rights to do so
- AMQP aims to be technology and language-neutral
  - Available in C, C++, Java, JMS, .NET, C#, Ruby, Python, etc.
  - Requires IP, and can be used with TCP, UDP, SCTP, Infiniband, etc.
- Products complying with AMQP are inter-operable
  - AMQP is a Wire-Level protocol based on the ubiquitous IP
  - Wire-level compatibility means it can be embedded in the network
  - Applications written to Product X will plug into servers running Product Y
- Red Hat is a founding member of the AMQP Working Group

MRG: Realtime

- Enables applications and transactions to run predictably, with guaranteed response times
    - Provides microsecond accuracy
- Provides competitive advantage & meets SLA's
    - Travel web site: missed booking
    - Program trading: missed trades
    - Command & Control: life & death
- Provides replacement kernel for RHEL 5.1+; x86/x86_64
- Preserves RHEL Application Compatibility



Detail zoom-in of RHEL5 vs MRG Realtime



MRG: Realtime Tools

- **MRG includes a new MRG Realtime Latency Tracer**
  - Runtime trace capture of longest latency codepaths – both kernel and application. Peak detector
  - Selectable triggers for threshold tracing
  - Detailed kernel profiles based on latency triggers
- **Existing standard RHEL5 based performance monitoring tools remain relevant**
  - Gdb, OProfile Frysk – source level debuggers & profiler
  - SystemTap, kprobe – kernel event tracing and dynamic data collection
  - kexec/kdump standard kernel dump/save core capabilities

MRG: Realtime Breakthroughs

- **Red Hat engineers succeed at mainstream acceptance**
  - Methodical multi-year implementation
  - Incrementally added features beneficial to all use cases
  - Iteratively worked to cooperatively build an inclusive realtime community
- **MRG Realtime is COTS (Commercial Off The Shelf) operating system**
  - Standard RHEL5.1 OS
  - Replacement kernel
- **Integrated with distributed high speed messaging & grid scheduler**
  - MRG Messaging, MRG Grid

MRG: Grid

- Brings advantages of scale-out and flexible deployment to any application
- Delivers better asset utilization, allowing applications to to take advantage of all available computing resources
- Dynamically provisions additional peak capacity for “Christmas Rush”-like situations
- Executes across multiple platforms and in virtual machines
- Provides seamless and flexible High Throughput Computing (HTC) and High Performance Computing (HPC) across
  - Local grids
  - Remote grids
  - Remote clouds (Amazon EC2)
  - Cycle-stealing from desktop PCs

MRG: Grid based off Condor

- MRG Grid is based on the Condor Project created and hosted by the University of Wisconsin, Madison
- Red Hat and the University of Wisconsin have signed a strategic partnership around Condor:
  - University of Wisconsin makes Condor source code available under OSI-approved open source license
  - Red Hat & University of Wisconsin jointly fund and staff Condor development on-campus at the University of Wisconsin
- Red Hat and the University of Wisconsin's partnership will:
  - Add enhanced enterprise features, management, and supportability to Condor and MRG Grid
  - Add High Throughput Computing capabilities to Linux

Red Hat Enterprise MRG Availability

- MRG Announcement & Beta Launch: December 2007
  - Public beta

- MRG v1.0: Early 2008
  - RHEL-only support for MRG Messaging broker
  - MRG Grid Technology Preview

- MRG v1.1: Late 2008
  - Multi-platform support for MRG Messaging Java-based broker
  - AMQP support updated to newly available AMQP version (1.0)
  - MRG Grid support available

Additional Information:

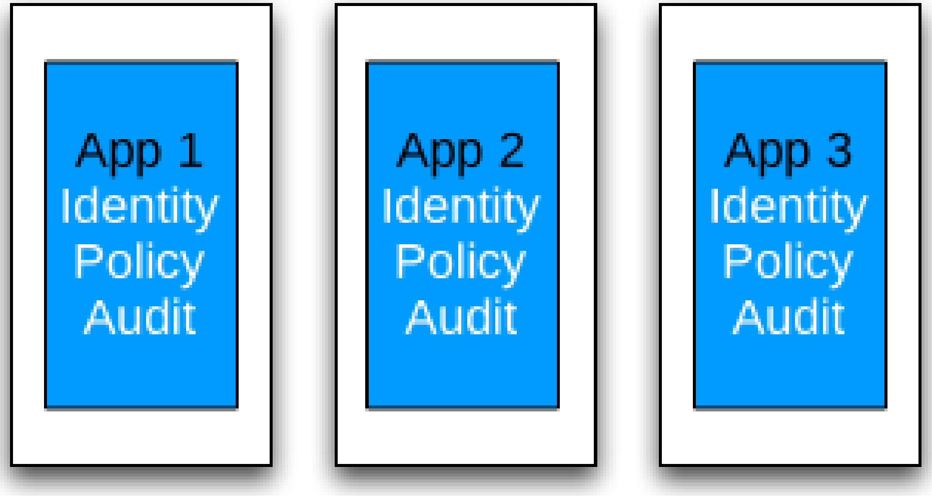
http://www.redhat.com/mrg/

Security Information Situation Today

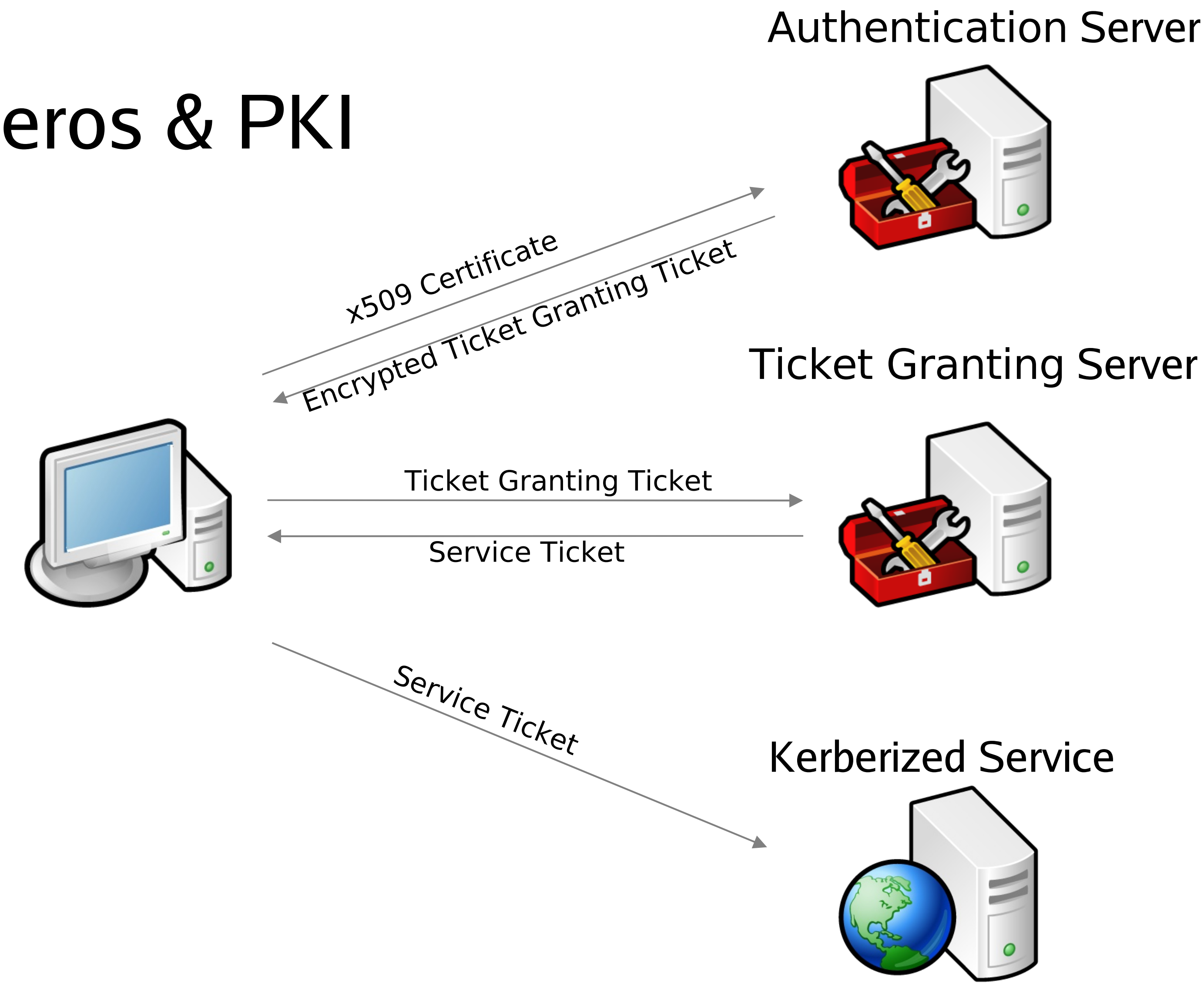
- Many security and security management applications store and manage their own vital security information
  - Identity
  - Policy
  - Audit

- Difficult to analyze across applications, so organizations can't
  - Form a full picture of their security stance
  - Comply with government regulations
  - Protect themselves sufficiently
  - Efficiently enable their operations

- Example: Identity silos



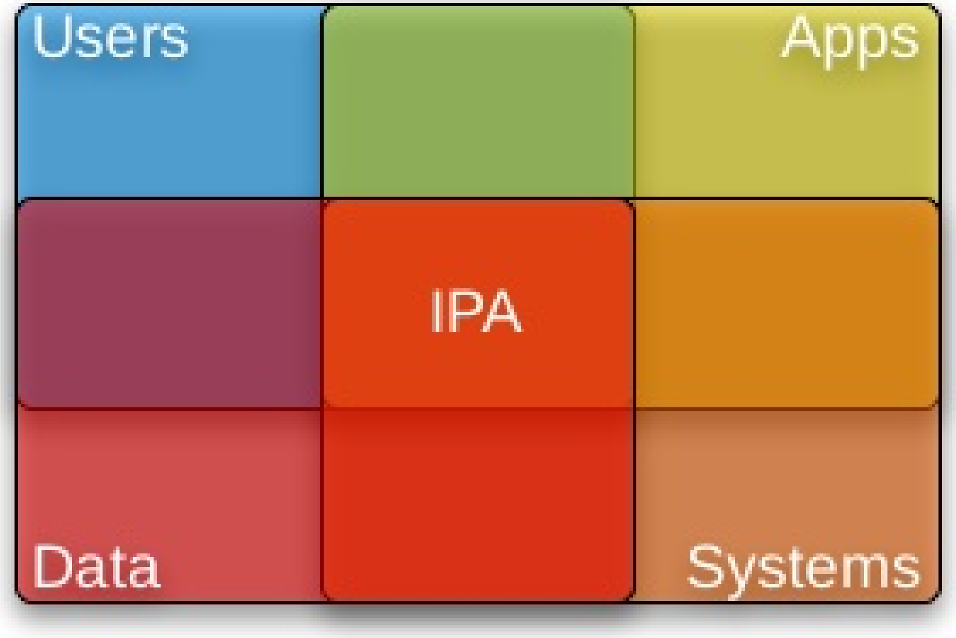
Kerberos & PKI



- Enterprise Single Sign-on
- Strong authentication with strong credentials

What is needed?

To enable this:



Maximize freedom

Maximize efficiency

Vital security information (IPA) should be:

- Open (You own it)
- Inter-operable
- Manageable

Need a way to make it possible for vital security information

- Identity
- Policy
- Audit

to enable the freedom and efficiency of next generation IT infrastructure

freeIPA

identity | policy | audit

- Project
  - Open Source
  - www.freeipa.org
  - Started and contributed to by Red Hat
  - Open to all
  - IPA = Identity, Policy, Audit

- Big vision
  - Start with centralized user identity management for UNIX/Linux
  - Add robust, shared sense of machine, service and data identity
  - Provide centrally managed admin access control for UNIX/Linux
  - Give ability to externalize policy and add to it easily
  - Add centralized audit
  - With this you can enable flexible cross-enterprise policy and rational audit

freeIPA
identity | policy | audit

IPAv1 (February target) will provide

Single Sign on for users

- Tie together Directory and Kerberos
- User Kerberos ticket for SS) to UNIX/Linux, JBoss, other apps

Centralized authentication point for IT

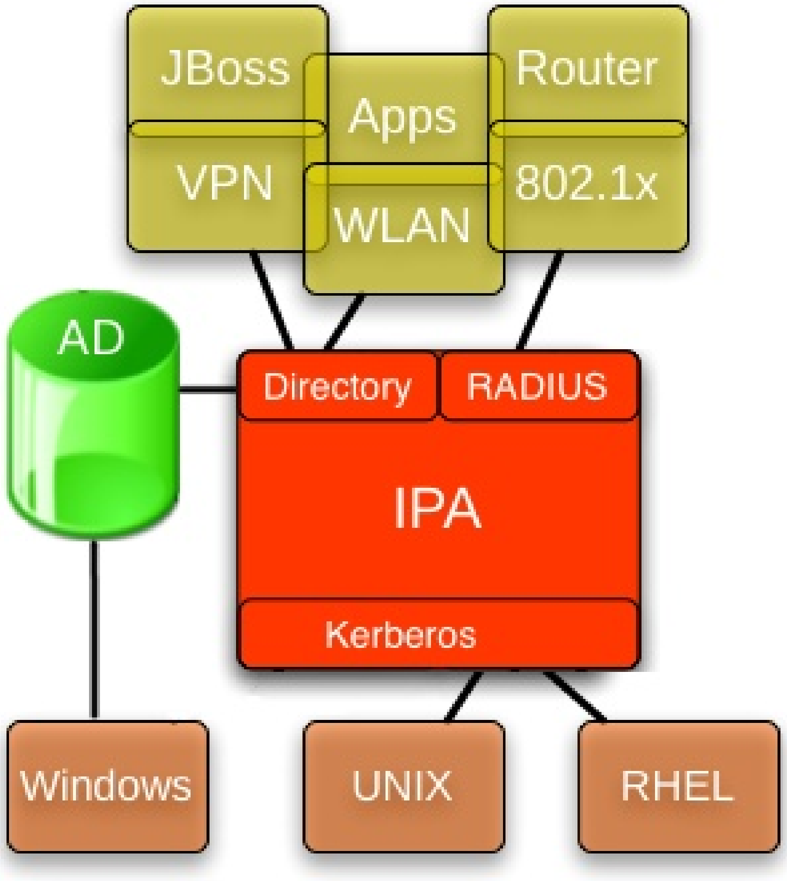
- Unite Directory, Kerberos, RADIUS servers, SAMBA
- From Apps, UNIX/Linux, VPNs, WLANs

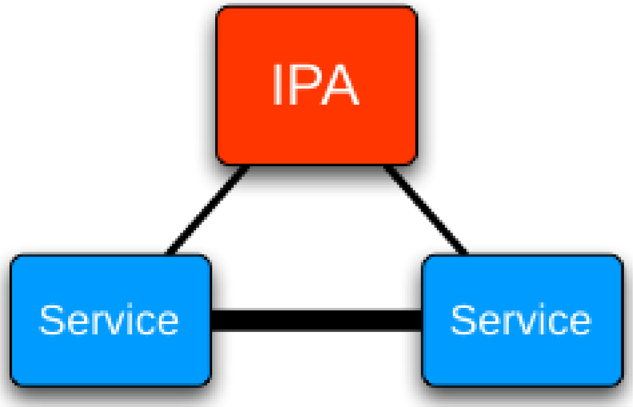
Easy for IT to set up, migrate to, and manage

- Simple IPA install
- Intuitive web interface, Command line
- Tools migrate from NIS

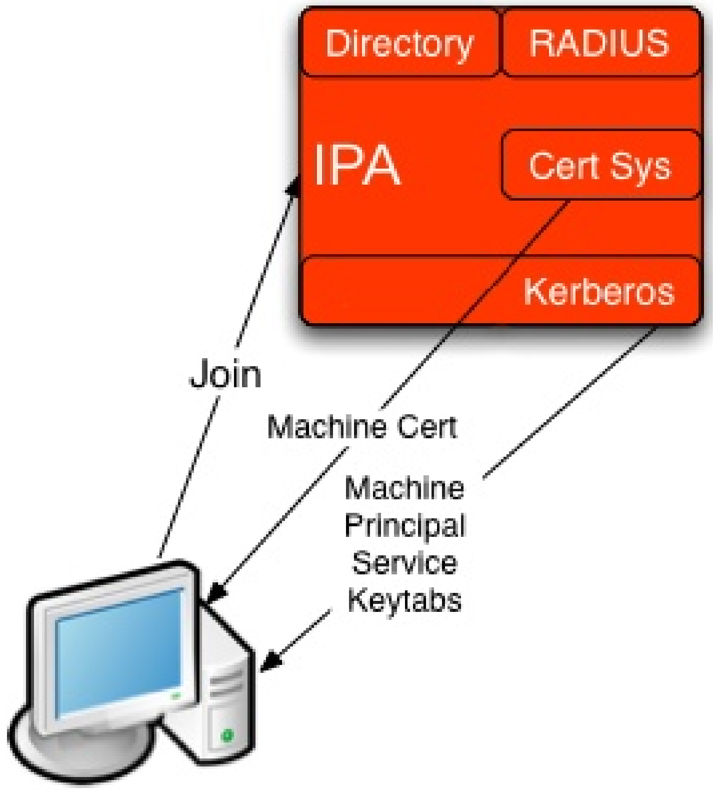
Key Data replicated via Directory

Process identity via a Kerberos principal





freeIPA
identity | policy | audit



IPAv2 (July target) will provide

Identify and group machines, Vms, services

Simplified service authentication and establishment of secure communication

- Machine identity via Kerberos, certificate
- Process identity via Kerberos principal

Management of machine certificate

Centrally managed access control

- Extensible policy framework
- Set policy of which users can access which apps on which machines
- Centrally managed scoped admin control

Central audit database

- Centrally audit security event, logs, keystrokes (?), compliance with lockdown

Open Discussion