Solaris™ 10 Security Overview

Glenn M. Brunette, Jr.
DE, Chief Security Architect
Client Solutions
Sun Microsystems, Inc.

Sun Developer Day III

Agenda

- Overview of Solaris 9 Security (brief)
- Solaris 10 Security Goals
- Key Solaris 10 Security Enhancements
- Additional Security Features
- References

Solaris 9 Security Overview

- Access Control Lists
- Role-based Access Control
- IPsec / IKE
- Solaris Auditing
- TCP Wrappers (inetd)
- Flexible Crypt
- Signed Patches
- Granular Packaging
- SSL-enabled LDAP
- WAN Boot
- IKE Hardware Accel.

- Solaris Fingerprint DB
- Solaris Secure Shell
- Kerberos
- /dev/[u]random
- Enhanced PAM Framework
- Smartcard Framework
- Java 1.4 Security
- SunScreen 3.2
- Solaris Security Toolkit
- sadmind DES Auth
- LDAP Password Management

Security Goals — Defensive

- Provide strong assurance of system integrity
  - Simplify building and deploying of secure solutions
  - Monitor system state for unexpected change
  - Audit security relevant changes
- Defend system from unauthorized access
  - Contain damage caused by unauthorized access
  - Minimize privileges given to people and processes
  - Filter inbound communications into the system

Security Goals — Enabling

- Secure authentication of all active subjects
  - Use strong user and host level authentication
  - Integrate authentication mechanisms
  - Leverage a unified authentication infrastructure
- Protect communications between endpoints
  - Provide private data transmissions
  - Verify integrity of received data
  - Securely establish and protect keys

Security Goals — Deployable

- Emphasize integratable stack architecture
  - Enable pluggable use of 3rd party security providers
  - Provide abstracted APIs for customers
  - Offer robust security platform for Sun's products
- Interoperable with other security architectures
- Ease management and use of security features
  - Transparently maintain security infrastructure
  - Simplify and centralize security policy definition
  - Minimize visibility of secure features to end users
- Receive independent assessment of security

Stronger “Out of the Box” Posture

- New Minimal Meta-Cluster (SUNWCrnet)
  - Solid foundation for minimizing systems.

    192M, 28 set-uid, 11 set-gid, 91 pkgs, 2 listening services
- New Hardened Service Profile
  - generic_limited_net
- More Conservative, Post-Install Posture
  - More services are “off” by default.
  - Stronger default security settings.
- Fortified Code Base
  - Benefit from continued security reviews.

    Not just for security bugs, but also to better contain privileges used by set-id and other programs!

Service Management Framework

- New model for service management.
- SMF benefits include:
  - Consistent service representation
  - Common set of management interfaces
  - Parallelized startup of services
  - Automatic dependency resolution
  - Delegated service restarts
- Simplifies disabling unused services.
  - Solaris Security Toolkit will use SMF in Solaris 10.
- Integrated with RBAC and Privileges
  - SMF Management, Service Start, etc.

SMF Example #1

```
# svcs network/inetd
STATE          STIME    FMRI
online         1:28:15 svc:/network/inetd:default
# svcadm disable network/inetd
# svcs network/inetd
STATE          STIME    FMRI
disabled        1:46:31 svc:/network/inetd:default
# svcs -x -v network/inetd
svc:/network/inetd:default (inetd)
 State: disabled since Wed Dec 01 01:46:31 2004
Reason: Disabled by an administrator.
   See: http://sun.com/msg/SMF-8000-05
   See: man -M /usr/share/man -s 1M inetd
Impact: 18 services are not running:
        svc:/network/rpc-100068_2-5/rpc_udp:default
        svc:/network/rpc/gss:ticotsord
        [...]
```

SMF Example #2

```
# svcprop -v -p defaults network/inetd
defaults/bind_addr astring ""
defaults/bind_fail_interval integer -1
defaults/bind_fail_max integer -1
defaults/con_rate_offline integer -1
[...]
defaults/stability astring Evolving
defaults/tcp_trace boolean false
defaults/tcp_wrappers boolean false

# svcs -x network/smtp
svc:/network/smtp:sendmail (sendmail SMTP mail transfer agent)
 State: maintenance since Wed Dec 01 01:31:35 2004
Reason: Start method failed repeatedly, last exited with status 208.
   See: http://sun.com/msg/SMF-8000-KS
   See: sendmail(1M)
Impact: 0 services are not running.
```

SMF Example #3

```
# svcprop -v -p start apache2
start/exec astring /lib/svc/method/http-apache2\ start
start/timeout_seconds count 60
start/type astring method
start/user astring webservd
start/group astring webservd
start/privileges astring basic,!proc_session,!proc_info,!file_link_any,net_privaddr
start/limit_privileges astring :default
start/use_profile boolean false
start/supp_groups astring :default
start/working_directory astring :default
start/project astring :default
start/resource_pool astring :default
```

User/Password Management

- Local Password Complexity Checks
  - Login Name, White Space
  - Mininum Alpha, Non-Alpha, Upper, Lower, (Consequtive) Repeats, Special, Digits, etc.
- Local Password History
  - 0 to 26 Passwords Deep.
- Local Banned Password List (Dictionary)
- Local Account Lockout (3 Strikes)
- New Password Command Options:
  - Non-Login, Locked and Unlocked

Secure Remote Access - Kerberos

- Kerberos Enhancements
  - MIT Kerberos 1.3.2 Refresh
  - KDC Incremental Propagation
  - kclient Auto-configuration Tool
  - pam_krb5_migrate KDC Auto-population Tool
  - TCP and IPv6 Support
  - AES-128, AES-256, 3DES, RC4-HMAC Support
  - SPNego – GSS-API Dynamic Security Negotiation
  - Bundled Remote Applications (Clients & Servers)

    telnet, ftp, rlogin, rsh, rcp, rdist, Secure Shell
    Mozilla and Apache
  - Interoperability Fixes

Secure Remote Access - SSH

- Secure Shell Enhancements
  - OpenSSH 3.6p2++ Refresh
  - GSS-API Support
  - Enahnced Password Aging Support
  - Keyboard “Break” Sequence Support
  - X11 Forwarding “on” by default
  - RC4, AES CTR mode Encryption Support
  - /etc/default/login Synchronization
  - SSH2 Rekeying
  - Server Side Keepalives

Process Privileges

- Execute with only those privileges that are actually needed.
  - Delegation of “root” authority.
  - Completely backward compatible.
  - Allows fine-grained control of privilege (nearly 50!)
  - Privileges are inheritable, relinquishable, etc.
- Check for privileges and not just UID == 0!
- Mitigate effects of future flaws.
  - Drop any privileges you do not need (or others once you are done with them).

Process Privileges Listing

contract_event
contract_observer
cpc_cpu
dtrace_kernel
dtrace_proc
dtrace_user
file_chown
file_chown_self
file_dac_execute
file_dac_read
file_dac_search
file_dac_write
file_link_any
file_owner
file_setid
ipc_dac_read
ipc_dac_write
ipc_owner
net_icmpaccess
net_privaddr
net_rawaccess
proc_audit
proc_chroot
proc_clock_highres
proc_exec
proc_fork
proc_info
proc_lock_memory
proc_owner
proc_priocntl
proc_session
proc_setid
proc_taskid
proc_zone
sys_acct
sys_admin
sys_audit
sys_config
sys_devices
sys_ipc_config
sys_linkdir
sys_mount
sys_net_config
sys_nfs
sys_res_config
sys_resource
sys_suser_compat
sys_time

Process Privilege Sets

- Effective Set
  - Privileges currently in effect
  - Privileges can be added or dropped
- Permitted Set
  - Upper bound on Effective Set for this process
  - Privileges can be dropped (changes Effective)
- Inheritable Set
  - Default privileges given to child processes
  - Becomes child's Permitted and Effective Set
- Limit Set
  - Upper bound for Inheritable Set
  - Typically contains all privileges

Process Privilege Inheritance

- Limit (L) is unchanged
- L is used to bound privs in Inheritable (I)
  - $I' = I \cap L$
- Child's Permitted (P') & Effective (E') are:
  - $P' = E' = I'$
- Typical process
  - $P = E = I = \{basic\}$
  - $L = \{all\ privileges\}$
  - Since P = E = I, children run with same privileges

Root Account Still Special

- root owns all configuration/system files
  - uid 0 is therefore still very powerful
- Privilege escalation prevention
  - Require ALL privileges to modify objects owned by root when euid ≠ 0
  - Fine tuning in certain policy routines

    Not all privileges ⇒ only nosuid mounts
- Prefer services be non-0 uid + privileges
  - Additive approach is safer than uid 0 – privileges

Using Process Privileges

- Four Primary Methods
  - ppriv(1)

    ```
    # ppriv -e -D -s -proc_fork,-proc_exec /bin/sh -c finger
    sh[387]: missing privilege "proc_fork" (euid = 0, syscall = 143) needed at cfork+0x18
    /bin/sh: permission denied
    ```

  - User Rights Management (RBAC)

    ```
    # grep “Network Management” /etc/security/exec_attr
    Network Management:solaris:cmd:::/sbin/ifconfig:privs=sys_net_config
    Network Management:solaris:cmd:::/sbin/route:privs=sys_net_config
    ```

  - Service Management Framework (SMF)

    ```
    # svcprop -p start system/cron | grep privileges
    start/privileges astring :default
    start/limit_privileges astring :default
    ```

  - Privilege Aware Applications

    Drop unneeded privileges, bracket privileged code, etc.

Process Privileges Example #1

```
# ppriv `pgrep rpcbind`
126:    /usr/sbin/rpcbind
flags = PRIV_AWARE
E: basic,!file_link_any,net_privaddr,!proc_exec,!proc_info,!proc_session,sys_nfs
I: basic,!file_link_any,!proc_exec,!proc_fork,!proc_info,!proc_session
P: basic,!file_link_any,net_privaddr,!proc_exec,!proc_info,!proc_session,sys_nfs
L: basic,!file_link_any,!proc_exec,!proc_fork,!proc_info,!proc_session
# ppriv $$
341:    -sh
flags = <none>
E: all
I: basic
P: all
L: all
```

Process Privileges Example #2

```
# ppriv -e -D -s -proc_fork,-proc_exec /bin/sh -c finger
sh[387]: missing privilege "proc_fork" (euid = 0, syscall = 143) needed at cfork+0x18
/bin/sh: permission denied

# touch /foo
# chown bin /foo
# chmod 600 /foo
# cat /foo
# ppriv -e -D -s -file_dac_read cat /foo
cat[393]: missing privilege "file_dac_read" (euid = 0, syscall = 225) needed at
ufs_access+0x3c
cat: cannot open /foo

# ppriv -e -s -file_dac_read /bin/sh
# truss -f -vall -wall -tall cat /foo
[...]
397:    open64("/foo", O_RDONLY)                    Err#13 EACCES [file_dac_read]
[...]
```

Solaris Privileges Example #3

Solaris 9 Network Management Rights Profile

```
# grep "Network Management" /etc/security/exec_attr
Network Management:suser:cmd:::/usr/sbin/ifconfig:uid=0
Network Management:suser:cmd:::/usr/sbin/route:uid=0
[...]
```

Solaris 10 Network Management Rights Profile

```
# grep “Network Management” /etc/security/exec_attr
Network Management:solaris:cmd:::/sbin/ifconfig:privs=sys_net_config
Network Management:solaris:cmd:::/sbin/route:privs=sys_net_config
[...]
```

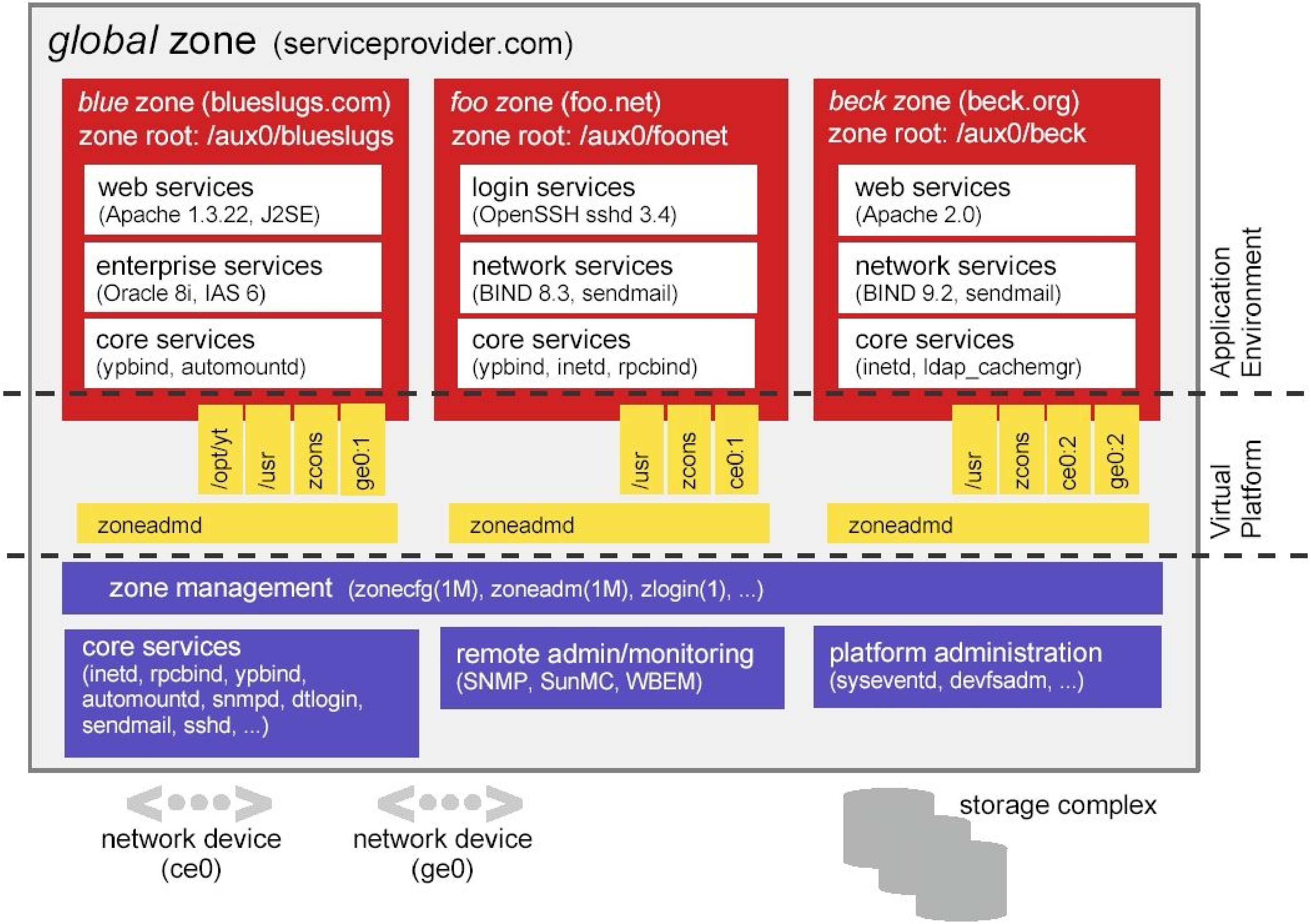
Solaris 10 Custom (BART) Rights Profile

```
# grep "^File Integrity:" /etc/security/exec_attr
File Integrity:solaris:cmd:::/usr/bin/bart:privs=file_dac_read,file_dac_search
```

Containers (aka Zones)

- Containers Overview
  - Containers are virtualized application environments.
  - Thousands of containers can be installed on a system.
  - Each acts like a separate operating system.
  - Each is in fact running on the same kernel.
- Containers Security Overview
  - Containers have security boundaries around them.
  - Containers operate with fewer privileges.
  - Important name spaces are isolated.
  - Processes running in a zone cannot affect other zones.
  - Cross-zone communication via network only (default).
  - Resources within a zone are strictly controlled.

Container Example



global zone (serviceprovider.com)

blue zone (blueslugs.com)
zone root: /aux0/blueslugs

web services
(Apache 1.3.22, J2SE)

enterprise services
(Oracle 8i, IAS 6)

core services
(ypbind, automountd)

foo zone (foo.net)
zone root: /aux0/foonet

login services
(OpenSSH sshd 3.4)

network services
(BIND 8.3, sendmail)

core services
(ypbind, inetd, rpcbind)

beck zone (beck.org)
zone root: /aux0/beck

web services
(Apache 2.0)

network services
(BIND 9.2, sendmail)

core services
(inetd, ldap_cachemgr)

zone management (zonecfg(1M), zoneadm(1M), zlogin(1), ...)

core services
(inetd, rpcbind, ypbind, automountd, snmpd, dtlogin, sendmail, sshd, ...)

remote admin/monitoring
(SNMP, SunMC, WBEM)

platform administration
(syseventd, devfsadm, ...)

Container Security

- By default, global zone “root” can see and do everything.
- Local zones are restricted in order to protect the security of the system:
  - System Calls
  - Device Manipulation
  - Privileges
  - System Resources

Container Security – System Calls

- Permitted System Calls:
  - chmod(2), chroot(2), chown(2), and setuid(2)
- Prohibited System Calls:
  - memcntl(2), mknod(2), stime(2), and pset_create(2)
- Limited System Calls:
  - kill(2)

Container Security – Devices

- /dev Permitted System Calls:
  - chmod(2), chown(2), and chgrp(1)
- /dev Prohibited System Calls:
  - rename(2), unlink(2), symlink(2), link(2), creat(2), and mknod(2)
- Forced nodevices mount option
  - Prevents import of malicious device files from NFS and other foreign sources.
- Security audit performed on all drivers included in default zone configuration.

Container Security – Privileges

- contract_event
- contract_observer
- cpc_cpu
- dtrace_kernel
- dtrace_proc
- dtrace_user
- file_chown
- file_chown_self
- file_dac_execute
- file_dac_read
- file_dac_search
- file_dac_write
- file_link_any
- file_owner
- file_setid
- ipc_dac_read
- ipc_dac_write
- ipc_owner
- net_icmpaccess
- net_privaddr
- net_rawaccess
- proc_audit
- proc_chroot
- proc_clock_highres
- proc_exec
- proc_fork
- proc_info
- proc_lock_memory
- proc_owner
- proc_priocntl
- proc_session
- proc_setid
- proc_taskid
- proc_zone
- sys_acct
- sys_admin
- sys_audit
- sys_config
- sys_devices
- sys_ipc_config
- sys_linkdir
- sys_mount
- sys_net_config
- sys_nfs
- sys_res_config
- sys_resource
- sys_suser_compat
- sys_time

Container Example #1

```
# zonecfg -z myzone
myzone: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:myzone> create
zonecfg:myzone> set zonepath=/export/home/myzone
zonecfg:myzone> verify
zonecfg:myzone> commit
zonecfg:myzone>
# zoneadm -z myzone install
Preparing to install zone <myzone>.
Creating list of files to copy from the global zone.
Copying <2338> files to the zone.
Initializing zone product registry.
Determining zone package initialization order.
Preparing to initialize <803> packages on the zone.
Initialized <803> packages on zone.
Zone <myzone> is initialized.
The file </export/home/myzone/root/var/sadm/system/logs/install_log> contains a
log of the zone installation.
```

Container Example #2

```
# zoneadm -z myzone boot
# zoneadm list -v
  ID NAME             STATUS         PATH
   0 global           running        /
   2 myzone           running        /export/home/myzone

# zlogin -C myzone
[...]

# prtconf
System Configuration:  Sun Microsystems  sun4u
Memory size: 256 Megabytes
System Peripherals (Software Nodes):
prtconf: devinfo facility not available

# prtdiag
prtdiag can only be run in the global zone

# ppriv -D -e ifconfig hme0:1 down
ifconfig[9014]: missing privilege "sys_net_config" (euid = 0, syscall = 54) needed at
ip_sioctl_copyin_setup+0x108
ifconfig: setifflags: SIOCSLIFFLAGS: hme0:1: permission denied
```

Why run services in containers?

- Restricted Operations for Enhanced Security
  - Accessing raw memory, Dtrace, promiscuous mode snooping, altering network interface and route information, manipulating kernel modules, altering system time, etc.
- Resource Control and Management
  - CPU, Memory, Disk, Networking, etc.
- Enforcement with Assurance
  - Sparse Root Zones, IP Filter, Restricted Mount, etc.
- Observability with Integrity
  - BART, Solaris Auditing, etc.

Basic Auditing and Reporting Tool

- File-level integrity validation tool.
  - Operates in either “create” or “compare” mode.
  - “rules” files define what should be evaluated and how.
  - “manifest” files contain the results.
- Flexible operational methods.
  - Allows “BART” input and output to be stored locally, piped to another process (transmission, compression, encryption, signing, etc.)
- Very small footprint (1 binary).
- Can evaluate all zones from the global zone.
- Can even automate and centralize collection using BART, RBAC, Privileges, and SSH!

BART Examples

- BART rules (bart_rules(4))

```
/usr/sbin
CHECK all
```

- BART manifest (bart_manifest(4))

```
/usr/sbin/acctadm F 28356 100555 user::r-x,group::r-x,mask:r-x,other:r-x 414f3bb4
0 2 ece9d92d00b0c13ed2d56580e3856df7
```

- BART Create Operation:

```
# bart create -r rules > manifest
# find /usr/lib/nis | bart create -I > manifest
```

- BART Compare Operation:

```
# bart compare ./manifestA ./manifestB
/usr/sbin/auditd:
  acl  control:user::r-x,group::r-x,mask:r-x,other:r-x  test:user::r-x,group::r-x,mask:r-x,other:rwx
  contents  control:28dd3a3af2fcc103f422993de5b162f3 test:28893a3af2fcc103f422993de5b162f3
```

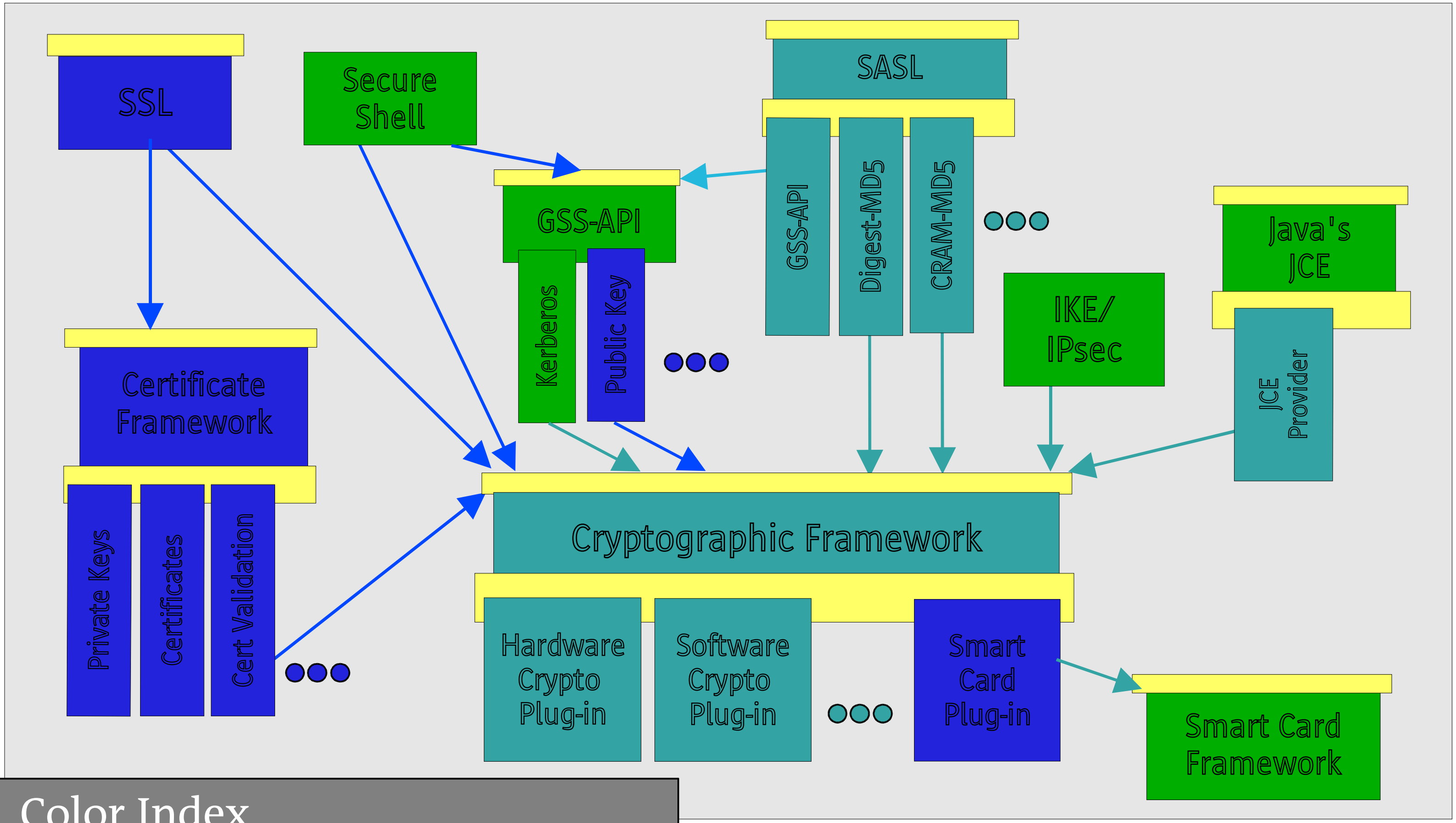
IP Filter

- Stateful and stateless packet inspection.
- Kernel-based packet filtering.
- Protocol proxies (TCP, UDP, FTP, rcmds, etc.)
- Text-based configuration.
- Support for both NAT and PAT.
- SYSLOG Logging.
- Small footprint, high performance.
- Minimal software requirements.

Cryptographic Framework

- Extensible cryptographic interfaces.
  - A common kernel and user-land framework for providing and using cryptographic functionality.
  - A common interface for cryptographic functions whether completed in hardware or software.
  - Extensible framework for vendors to provide custom functionality.
- By default, supports major algorithms.
  - Encryption: AES, RC4, DES, 3DES, RSA
  - Hashing: MD5, SHA-1
  - MAC: DES MAC, MD5 HMAC, SHA-1 HMAC
  - Optimized for both SPARC, Intel and AMD

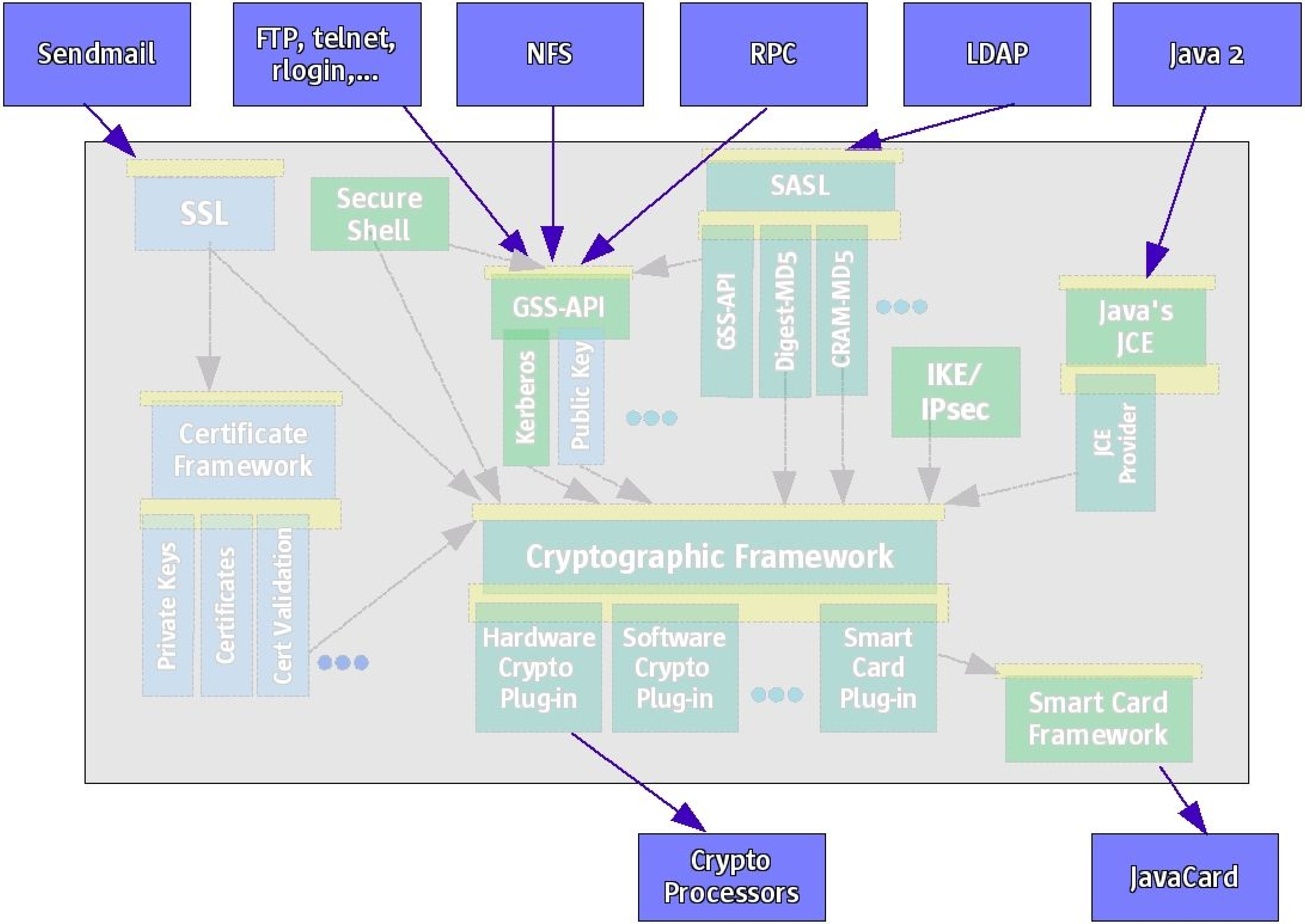
Cryptographic Framework



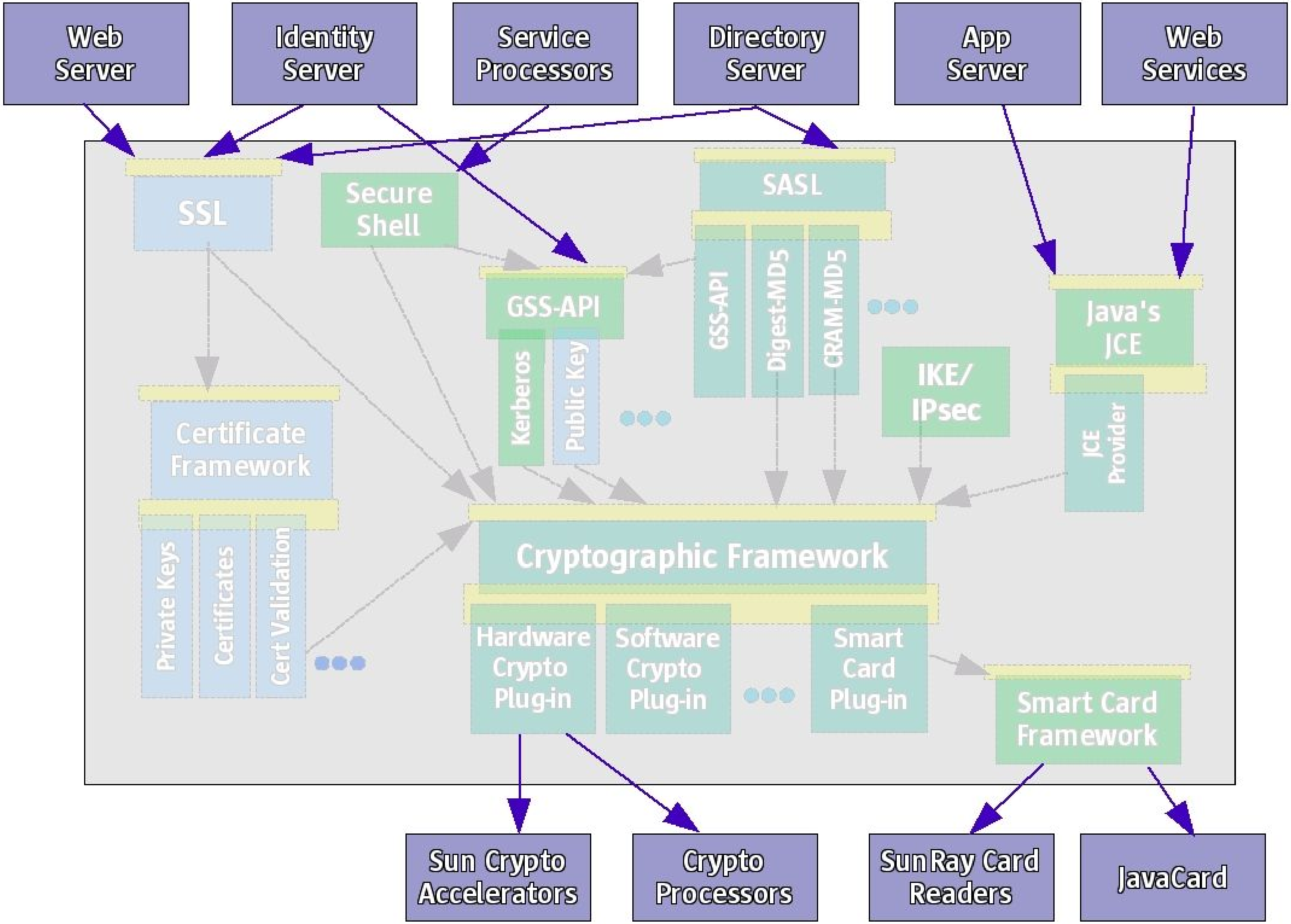
Color Index

- a Subsequent future release
- a Planned upcoming features
- a Available today (Solaris 9)
- 🟨 ISV Interfaces

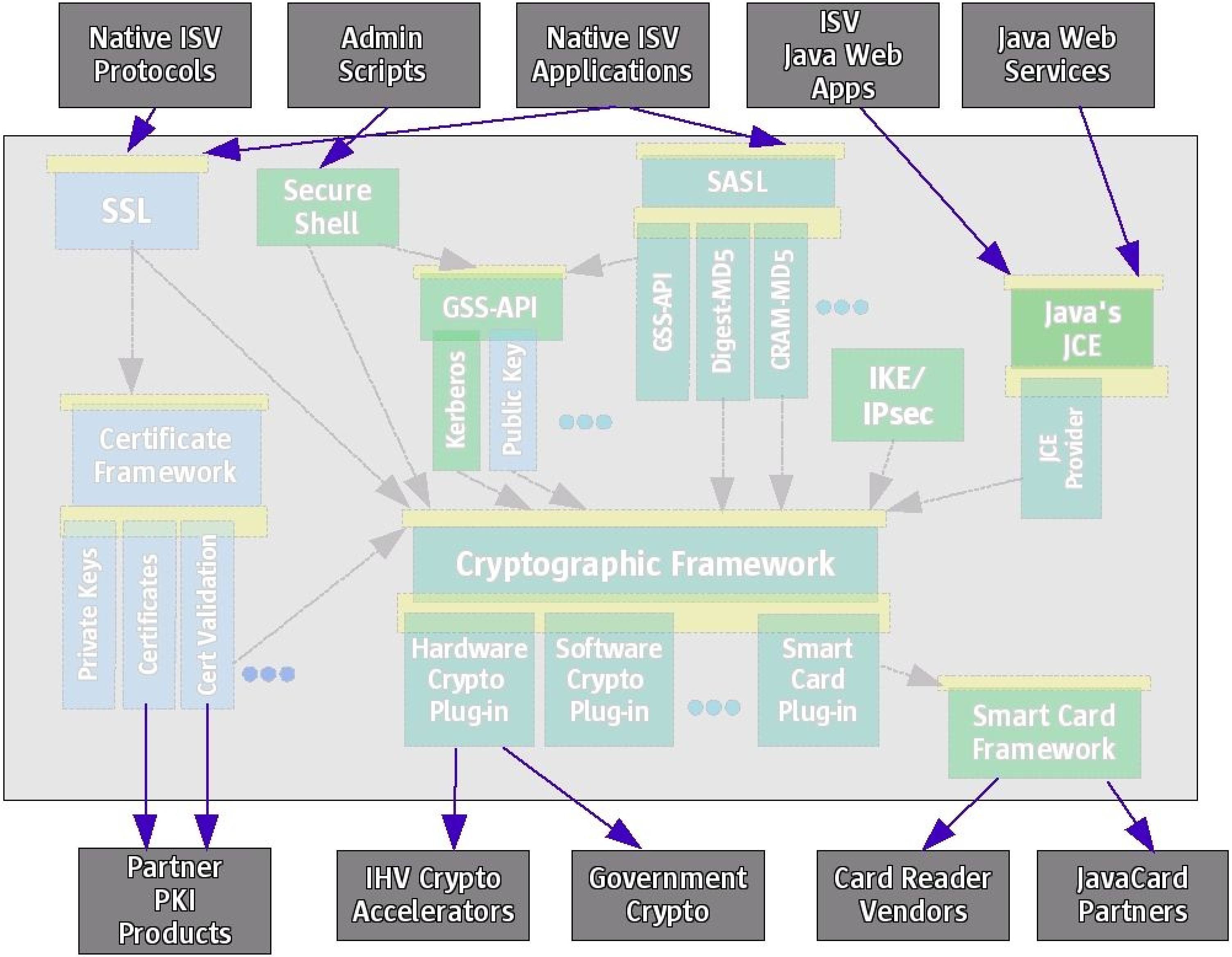
Security Platform for Solaris



Security Platform for Sun

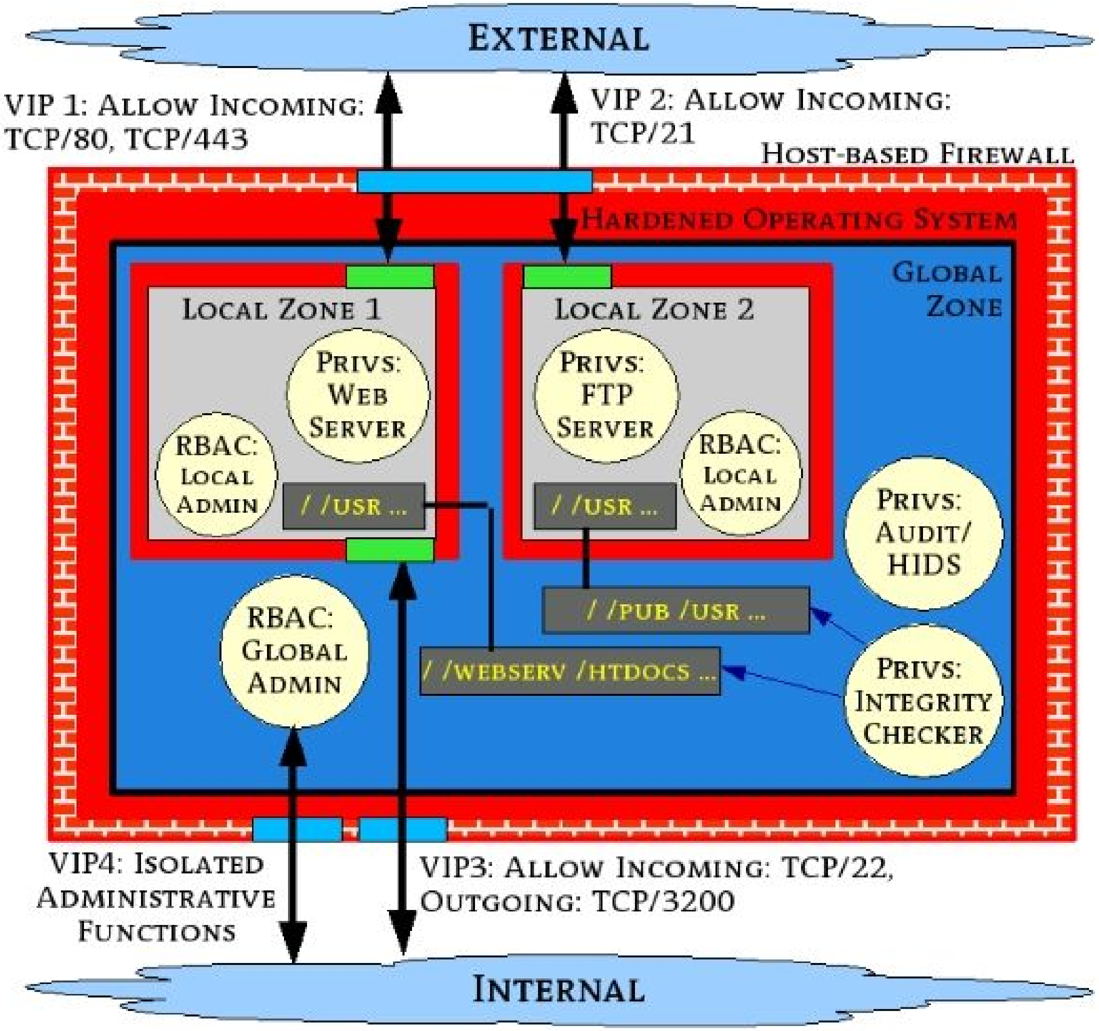


Security Platform for Partners

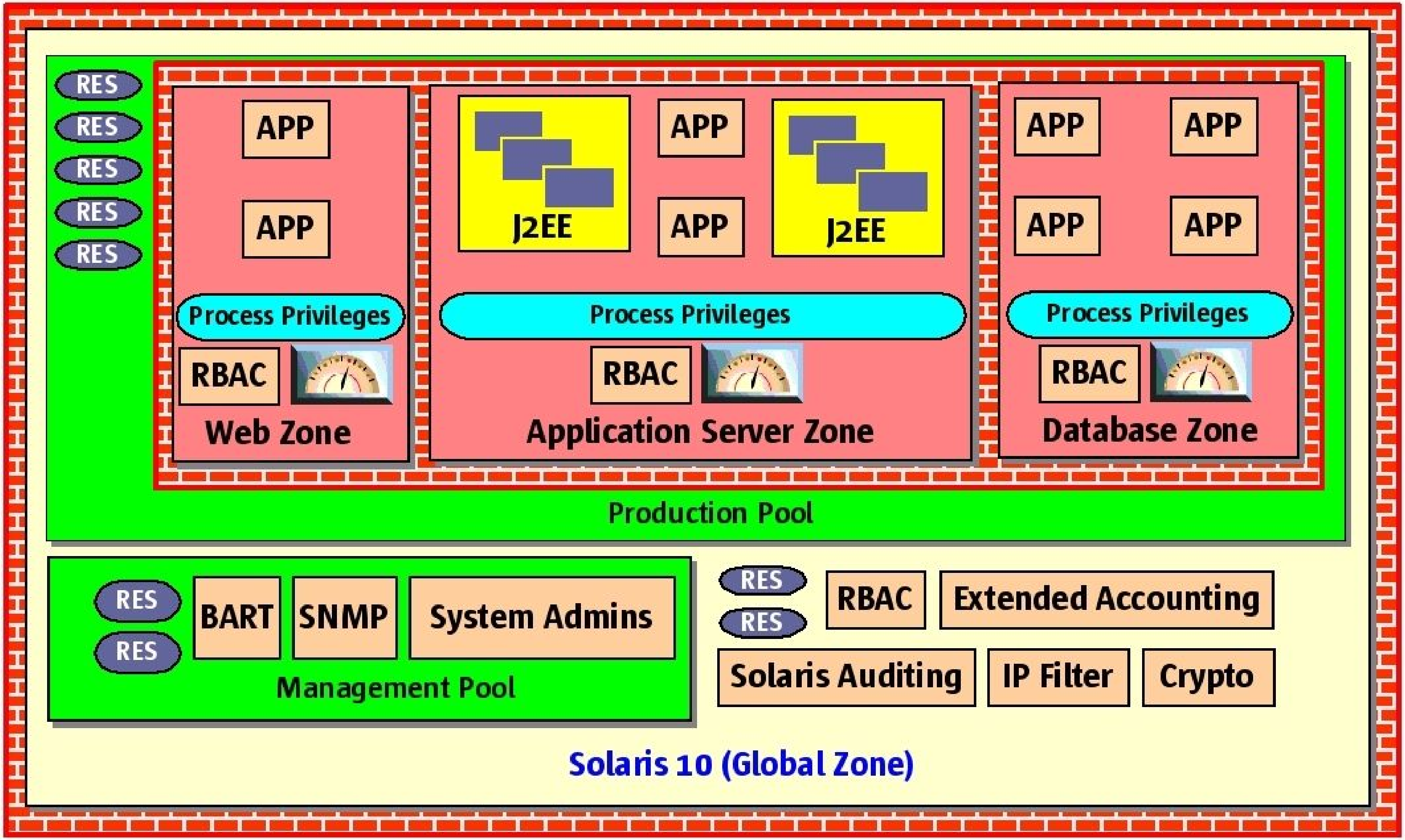


Putting it all together...

- Reduced Networking Meta Cluster for Minimization
- Solaris Security Toolkit
- Service Management Framework
- User Rights Management
- Process Rights Management
- Containers
- IP Filter
- Auditing
- Basic Auditing and Reporting Tool (BART)
- Cryptographic Framework
- Secure Remote Access and Administration



Putting it all together (2)



But wait! There's more!

- Auditing Improvements
  - Remote Logging via syslog
  - Audit Trail XML Translation
  - Audit Trail Noise Reduction
  - Audit Event Reclassification
- Enhanced TCP Wrappers Support
  - Now integrated with rpcbind and sendmail
- New Mount Options
  - noexec, nodevices
- User Process Visibility Restrictions
- vacation(1) Mail Filtering

and more...

- “root” GID is now “0” (root) not “1” (other)
- IPsec NAT Traversal
- RIPv2 Protocol Support
- ip_respond_to_timestamp now “0”.
- find(1) Support for ACLs
- “death by rm” safety
- OpenSSL libraries with a PKCS#11 engine
- Hardware RNG using Crypto Framework
- open(2) [O_NOFOLLOW], getpeerucred(3c), and many other developer enhancements...

and more...

- NFSv4
  - Support for GSS_API
- Sendmail 8.13
  - Support for rate limiting and milters.
- Java 1.5 Security
  - Security tokens, better support for more security standards (SASL, OCSP, TSP), various crypto and GSS security enhancements, etc.

... and the list keep right on going...

Summary

- Solaris security is very strong...
  - A 20 year history of continuous improvement.
  - Getting safer, simpler and better each day.
- Requested Actions:
  - Evaluate Solaris 10 Today!

    Try these new features and capabilities for yourself!
  - Consider a Solaris 10 Proof of Concept!

    Let us help you realize all of the benefits of the Solaris 10 OS (security and otherwise!)
  - Please Give Us Feedback!

    Tell us what you like, what you don't and where you think Solaris can be improved (and how)!

Solaris 10 Security Information

- Solaris 10 Home
  - http://www.sun.com/software/solaris/10/
- Solaris 10 Security Article
  - http://www.securityfocus.com/infocus/1776
- Solaris 10 Product Documentation
  - http://docs.sun.com/db/prod/solaris.10#hic
- Solaris 10 Security Blog Articles
  - http://blogs.sun.com/gbrunett
  - http://blogs.sun.com/casper
  - http://blogs.sun.com/arunpn
  - ... and many others...

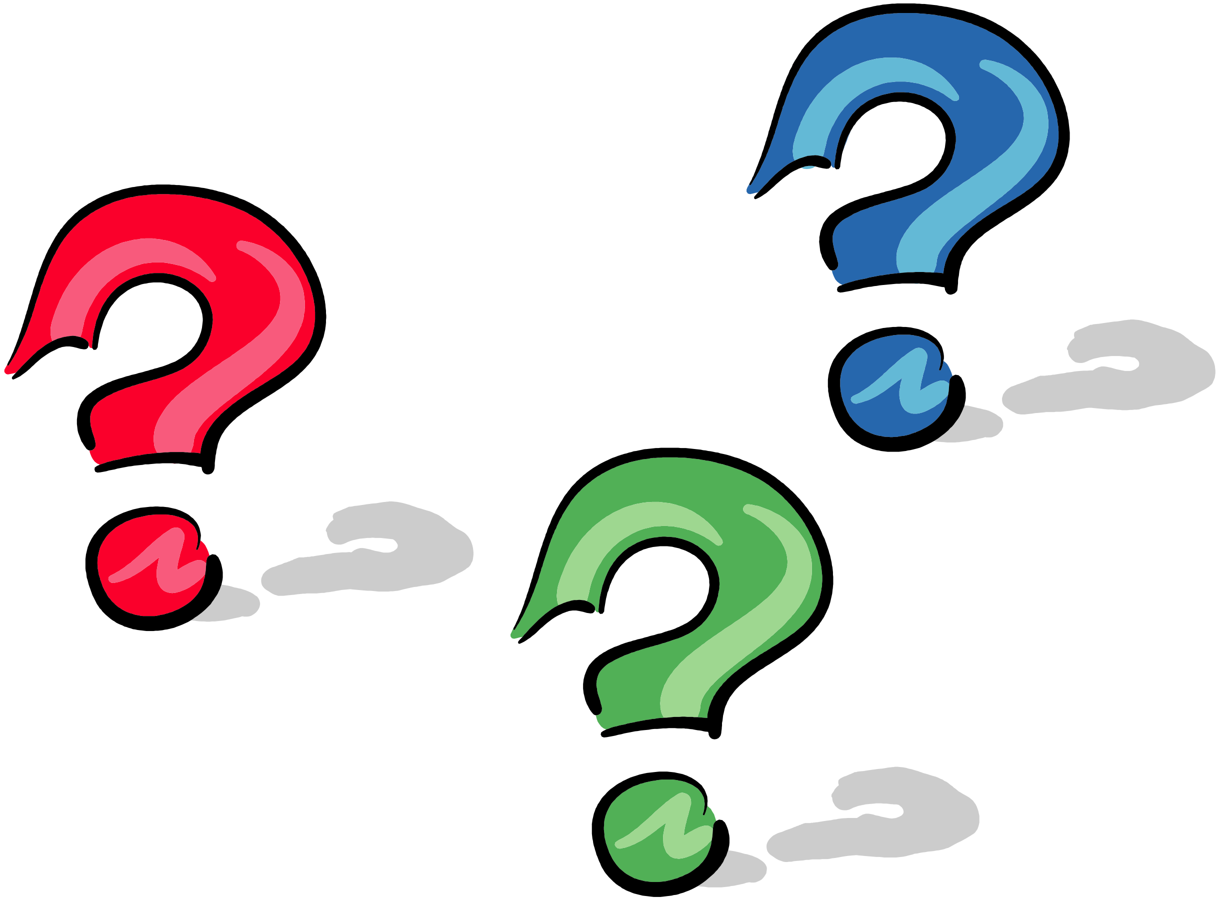
General Security Information

- Sun Security Home Page
  - http://www.sun.com/security/
- Solaris Patches & Finger Print Database
  - http://sunsolve.sun.com/
- Sun Security Coordination Team
  - http://sunsolve.sun.com/security/
- Sun BluePrints for Security
  - http://www.sun.com/security/blueprints/
- Solaris Security Toolkit
  - http://www.sun.com/security/jass/

Related Service Information

- Sun Client Solutions Security Services
  - http://www.sun.com/service/sunps/security
- Sun Education Security Services
  - http://suned.sun.com/US/catalog
- Sun Support Services
  - http://www.sun.com/service/support
- Sun Managed Security Services
  - http://www.sun.com/service/managedservices/

Questions?



Thank you!

glenn.brunette@sun.com