Bezpečnostní prvky OpenSolarisu

Martin Červený
M.Cerveny@sh.cvut.cz

Přehled bezpečnostních prvků OpenSolarisu

OpenSolaris Role Based Access Control

OpenSolaris kontejner / zóna

OpenSolaris Cryptographic Framework

Přehled bezpečnostních prvků OpenSolarisu

Cíle bezpečnosti

přehled

- otevřené standardy
  - POSIX, XPG, SUS, SVID, ANSI ...
- otevřená platforma
- otevřený kód
- ověřený kód
  - ***Common Criteria*** (ISO 15408)
    - CAPP EAL4+
    - RBACPP EAL4+
- AAA
  - Authentication – autentizace vstupu
  - Authorization – autorizace přístupu
  - Audit – účtování provozu

opensolaris <user group>

Autentizace vstupu

přehled

AAA

- konfigurovatelné jmenné služby pro identitu uživatele
  - ***Name Service Switch***
  - nsswitch.conf
  - moduly pro unix, LDAP, NIS, NIS+
  - typy služeb od passwd, ... RBAC, project
- konfigurovatelné ověření identity
  - ***Plugable Authentication Modules***
  - pam.conf
  - moduly pro unix, LDAP, NIS+, kerberos, čipové karty

Autorizace přístupu

přehled

AAA

- přístup k privilegovaným a administrativním nástrojům
  - ***Role Based Access Control***
- oprávnění pro přístup k privilegovaným funkcím kernelu
  - ***Privileges***
- soubory
  - ***POSIX Access Control List***
- sítě
  - ***IP filter***
  - ***TCP wrapper***
  - ***IPsec***
  - ***IP QoS***

opensolaris <user group>

Audit provozu

přehled

AAA

- audit v kernelu
  - ***Basic Security Module***
  - 257 událostí kernelu
  - 77 událostí aplikací
- informace o běhu aplikací
  - ***Extended Accounting***
- syslog
- souborový audit
  - ***Automated Security Enhancement Tool***
  - ***Basic Audit Reporting Tool***
- audit obsahu a bezpečnostní nastavení konfigurace
  - ***Solaris Security Toolkit (JASS)***

opensolaris™ <user group>

OpenSolaris Role Based Access Control

Cíle a komponenty

RBAC

- cíl
  - umožnit delegaci administrace systému
  - definice programových autorizací
  - snížit počet `setuid` programů
  - princip minimálních oprávnění
- komponenty RBAC
  - administrativní role (administrative roles)
  - programové autorizace (authorizations)
  - profily (profiles, rights), profily pro spuštění programů včetně bezpečnostních oprávnění kernelu (exec attributes, privileges)

opensolaris <user group>

Administrativní role

RBAC

komponenty

- standardní uživatelské konto, ale:
  - nelze se přímo přihlásit
  - aktivace pomocí ***su*** z autorizovaných kont
  - práce v ***Solaris Management Console***
  - použitý ***Profile Shell***
- konfigurace
  - roleadd(1m), rolemod(1m), roledel(1m)
  - user_attr(4) – doplňující soubor pro passwd(4) a shadow(4)

    user:::type=normal|**role**; roles=***users***;
    auths=***auths***; profiles=***profiles***;
    defaultpriv=***priv***; limitpriv=***priv***;
    project=***project***; lock_after_retries=yes|no

opensolaris <user group>

Programové autorizace

RBAC

komponenty

- hierarchický jmenný prostor

```
solaris.admin.printer.delete
solaris.admin.printer.modify
solaris.device.allocate
solaris.jobs.admin
solaris.jobs.grant
solaris.jobs.user
solaris.system.shutdown
...
```

- využití v aplikacích
  - getauthattr(3SECDB)
  - smc(1m), allocate(1m), crontab(1), at(1)...
- konfigurace
  - auth_attr(4)

    auth::***short descr***:***long descr***:help=***help***

Profily

RBAC

komponenty

1) profil definuje další atributy
   - prof_attr(4)
     profile:::***descr***:help=***help***;
     auths=***auths***; profiles=***profiles***; privs=***privs***
2) profil přiřazuje privilegia pro spouštění programů
   - implementace pomocí ***Profile Shell***
   - použití místo exec(2) transparentní externí spuštění setuid pfexec(1)
   - exec_attr(4)
     profile:suser|solaris:cmd:::***prog***:
     euid=***uid***; uid=***uid***; egid=***gid***; gid=***gid***;
     privs=***privs***; limitprivs=***privs***

Příklad profilu

RBAC

```
passwd
   jack:x:100:1:InsideJack:/home/jack:/bin/sh
   killer:x:200:1:Killer:/home/killer:/bin/pfsh
user_attr
   jack:::type=normal;roles=killer
   killer:::type=role;profiles=Killer
prof_attr
   Killer:::Process killer:
exec_attr
   Killer:suser:cmd:::/usr/bin/kill:euid=0
```

Privilegia

RBAC

- bezpečnostní oprávnění pro kernel, která nahrazují uid==0
- 48 privilegií ve skupinách
  - FILE – oprávnění přístupu k souborům
  - IPC – oprávnění přístupu k sdílené paměti, frontám zpráv a semaforům
  - NET – oprávnění přístupu k síti
  - PROC – oprávnění přístupu k procesům
  - SYS – oprávnění přístupu k systémovým zdrojům
- součástí informací kernelu o procesu
- nastavení privilegií
  - RBAC
  - privilegované programy (setuid)
  - v rámci startovací sekvence

opensolaris <user group>

Přehled privilegií

RBAC

privilegia

"contract_event" Process/Request critical/reliable events
"contract_observer" Obsever events other than euid
"cpc_cpu" Access to per-CPU perf counters
"dtrace_kernel" DTrace kernel tracing
"dtrace_proc" DTrace process-level tracing
"dtrace_user" DTrace user-level tracing
"file_chown" Change file's owner/group IDs
"file_chown_self" Give away (chown) files
"file_dac_execute" Override file's execute perms
"file_dac_read" Override file's read perms
"file_dac_search" Override dir's search perms
"file_dac_write" Override (non-root) file's write perms
"file_link_any" Create hard links to diff uid files
"file_owner" Non-owner can do misc owner ops
"file_setid" Set uid/gid (non-root) to diff id
"ipc_dac_read" Override read on IPC, Shared Mem perms
"ipc_dac_write" Override write on IPC, Shared Mem perms
"ipc_owner" Override set perms/owner on IPC
"net_icmpaccess" Send/Receive ICMP packets
"net_privaddr" Bind to privilege port (<1023+extras)
"net_rawaccess" Raw access to IP
"proc_audit" Generate audit records
"proc_chroot" Change root (chroot)
"proc_clock_highres" Allow use of hi-res timers
"proc_exec" Allow use of execve()
"proc_fork" Allow use of fork*() calls
"proc_info" Examine /proc of other processes
"proc_lock_memory" Lock pages in physical memory
"proc_owner" See/modify other process states
"proc_priocntl" Increase priority/sched class
"proc_session" Signal/trace other session process
"proc_setid" Set process UID
"proc_taskid" Assign new task ID
"proc_zone" Signal/trace processes in other zones
"sys_acct" Manage accounting system (acct)
"sys_admin System admin tasks (node/domain name)
"sys_audit" Control audit system
"sys_config" Manage swap
"sys_devices" Override device restricts (exclusive)
"sys_ipc_config" Increase IPC queue
"sys_linkdir" Link/unlink directories
"sys_mount" Filesystem admin (mount,quota)
"sys_net_config" Config net interfaces, routes, stack
"sys_nfs" Bind NFS ports and use syscalls
"sys_res_config" Admin processor sets, res pools
"sys_resource" Modify res limits (rlimit)
"sys_suser_compat" 3rd party modules use of suser
"sys_time" Change system time

opensolaris <user group>

Privilegia procesu

RBAC

privilegia

- informace kernelu o procesu
  - ***Effektive set (E)*** – aktuální oprávnění, dají se přidávat a odebírat shora omezené podle (P)
  - ***Permitted set (P)*** – horní omezení pro oprávnění (E) a (I), dají se pouze odebírat
  - ***Inheritable set (I)*** – nastavení výchozích oprávnění (nové E a P) pro synovský proces
  - ***Limited set (L)*** – horní omezení pro dědičná oprávnění (I) a nemůže nikdy růst
  - příznak ***PRIV_AWARE***
- spouštění setuid
  - pro nepřeprogramované (nejsou ***PRIV_AWARE***) procesy se v kernelu ověřuje jiné E a P

  E' = (euid == 0 ? L : E)

  P' = ((euid == 0 || ruid == 0 || suid == 0) ? L : P))

opensolaris <user group>

Nastavení privilegií

RBAC

privilegia

- RBAC
- nástroj ppriv(1)
  - zjištění privilegií u procesu
  - nastavení privilegií pro spouštěný proces
  - testování potřebných privilegií
- nástroj truss(1)
  - testování potřebných privilegií
- nastavení při startu OS
  - ***Service Management Facility***
  - definice v popisu (manifestu) aplikace
  - zobrazení nástrojem svcprop(1)
- API
  - set/getpflags(2), set/getppriv(2)

opensolaris <user group>

DEMO >>>>

OpenSolaris kontejner / zóna

Principy oddělení běhu aplikací

kontejner

- důvody
  - bezpečnostní oddělení
  - rozdělení nebo vyhrazení zdrojů
  - redundance a vysoká dostupnost
  - horizontální škálovatelnost
- možnosti dokonalého oddělení
  - oddělené systémy
  - dynamické hw rozdělení výkonných počítačů (hardware partitioning)
  - virtuální (emulovaný) hardware (virtual hardware monitors)
  - rozdělení na úrovni operačního systému (OS virtualization)

opensolaris <user group>

Kontejner OpenSolarisu

kontejner

- kontejner
  - rozdělení na úrovni operačního systému (zóny)
    - nezávislý běh instance operačního systému od úrovně init (sdílený kernel)
    - izolace softwarových chyb
    - bezpečnostní oddělení (méně privilegií)
  - vytvoření zdrojů podle procesorů (dynamic resource pools, processor sets)
  - vytvoření zdrojů podle výkonu procesorů na nižší úrovni (resource/workload manager, FSS scheduler (cpu-shares), projects)
  - vytvoření zdrojů podle řízení toků na IP úrovni (IP QoS)

opensolaris <user group>

Zóna OpenSolarisu

kontejner

zóna

- konfigurace – zones(5)
  - zonecfg(1m)
    - jméno
    - kořenový adresář zóny
    - souborové systémy
    - síťové rozhraní
    - další zařízení
    - řízení zdrojů
- instalace, spuštění a vypnutí
  - zoneadm(1m)
  - stav zóny
    - configured, incomplete, installed, ready, running, shutting_down, down
- zpřístupnění
  - zlogin(1), zonename(1)

opensolaris <user group>

Příklad

kontejner

zóna

```
zonecfg -z zone1
   zonecfg:zone1> create
   zonecfg:zone1> set zonepath=/zone1
   zonecfg:zone1> set autoboot=true
   zonecfg:zone1> add net
   zonecfg:zone1:net> set address=192.168.1.100
   zonecfg:zone1:net> set physical=gani0
   zonecfg:zone1:net> end
   zonecfg:zone1 > add fs
   zonecfg:zone1:fs> set dir=/usr/local
   zonecfg:zone1:fs> set special=/zone1/local
   zonecfg:zone1:fs> set type=lofs
   zonecfg:zone1:fs> end
   zonecfg:zone1> add device
   zonecfg:zone1:device> set match=/dev/ecpp0
   zonecfg:zone1:device> end
   zonecfg:zone1> verify
   zonecfg:zone1> commit
zoneadm -z zone1 install
zoneadm -z zone1 boot
zlogin -C zone1
```

opensolaris <user group>

Další vlastnosti zóny

kontejner

zóna

- žádné změny v API
- citlivé části souborového systému jen pro čtení
- omezení na zařízeních
- odebraná privilegia
  - cpc_cpu, dtrace_kernel, dtrace_proc, dtrace_user, gart_access, gart_map, net_rawaccess, proc_clock_highres, proc_lock_memory, proc_priocntl, proc_zone, sys_config, sys_devices, sys_ipc_config, sys_linkdir, sys_net_config, sys_res_config, sys_suser_compat, sys_time
- upravené nástroje
  - ps(1), prstat(1m), truss(1), iostat(1m), mpstat (1m), vmstat(1m), psrinfo(1m), sar(1)...
- upravená prezentace z kernelu
  - sysconf(3c), getloadavg(3c), kstat(3kstat), autofs, fd(4), mnttab(4), kill(2), SysV IPC, streams ...
- instalace aplikací
  - pkgmap(4), pkginfo(4)

opensolaris™ <user group>

Řízení zdrojů

kontejner

zdroje

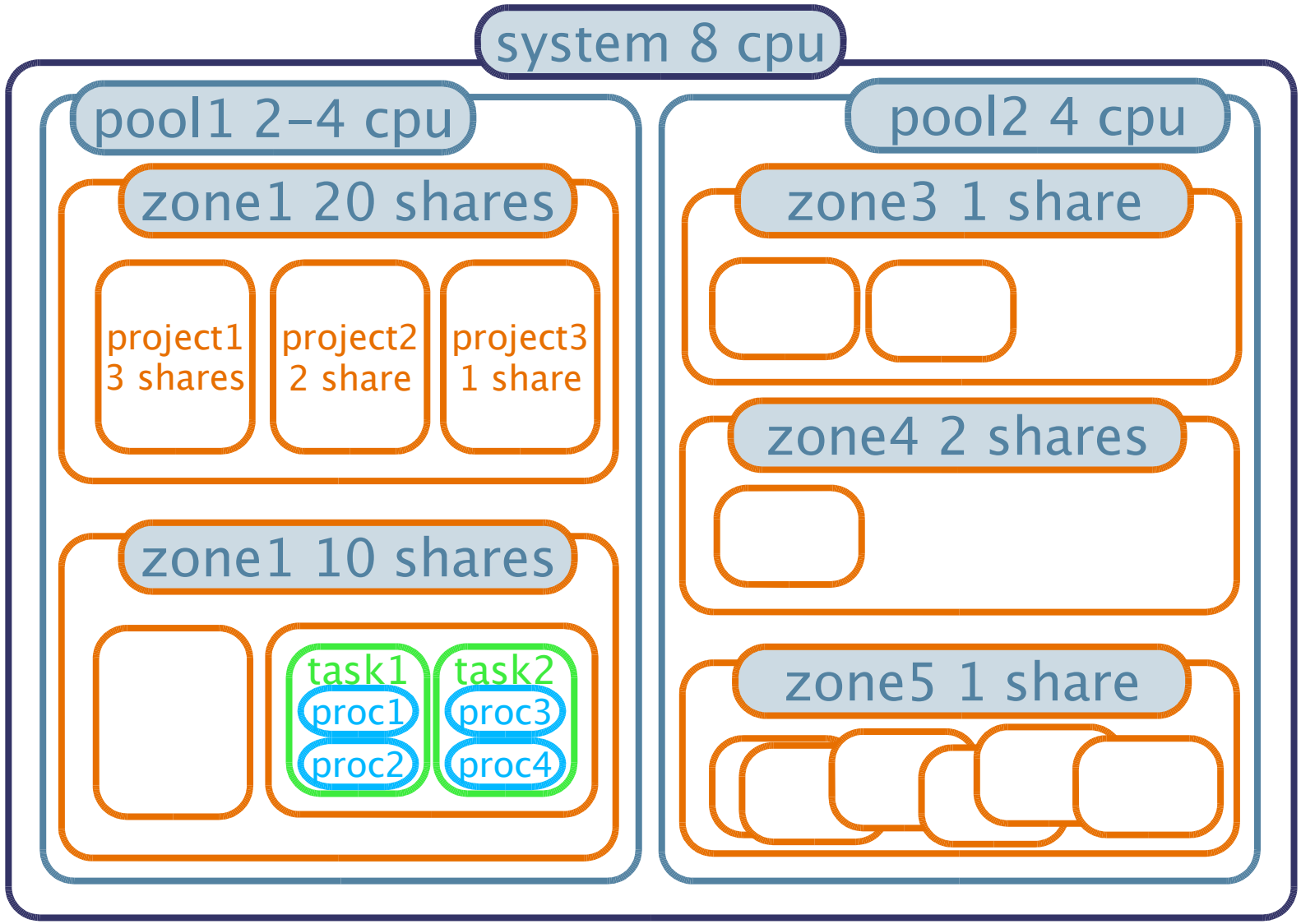
- konfigurace – ***dynamic resource pool***
  - pooladm(1m), poolcfg(1m), poolbind(1m), poolstat(1m), poold(1m)
  - pbind(1m), psrinfo(1m), psradm(1m), psradm(1m)
- konfigurace – ***FSS(7)***
  - resource_controls(5)
  - zonecfg(1m)
    - zones.cpu-shares
  - rctladm(1m)
  - prctl(1)
  - projadd(1m), projmod(1m), projdel(1m), projects(1), project(4), newtask(1)
    - project.cpu-shares

opensolaris <user group>

Rozdělení zdrojů

kontejner

zdroje



opensolaris
<user group>

kontejner

DEMO >>>>

opensolaris
<user group>

OpenSolaris Cryptographic Framework

Vlastnosti krypto. systému

krypto

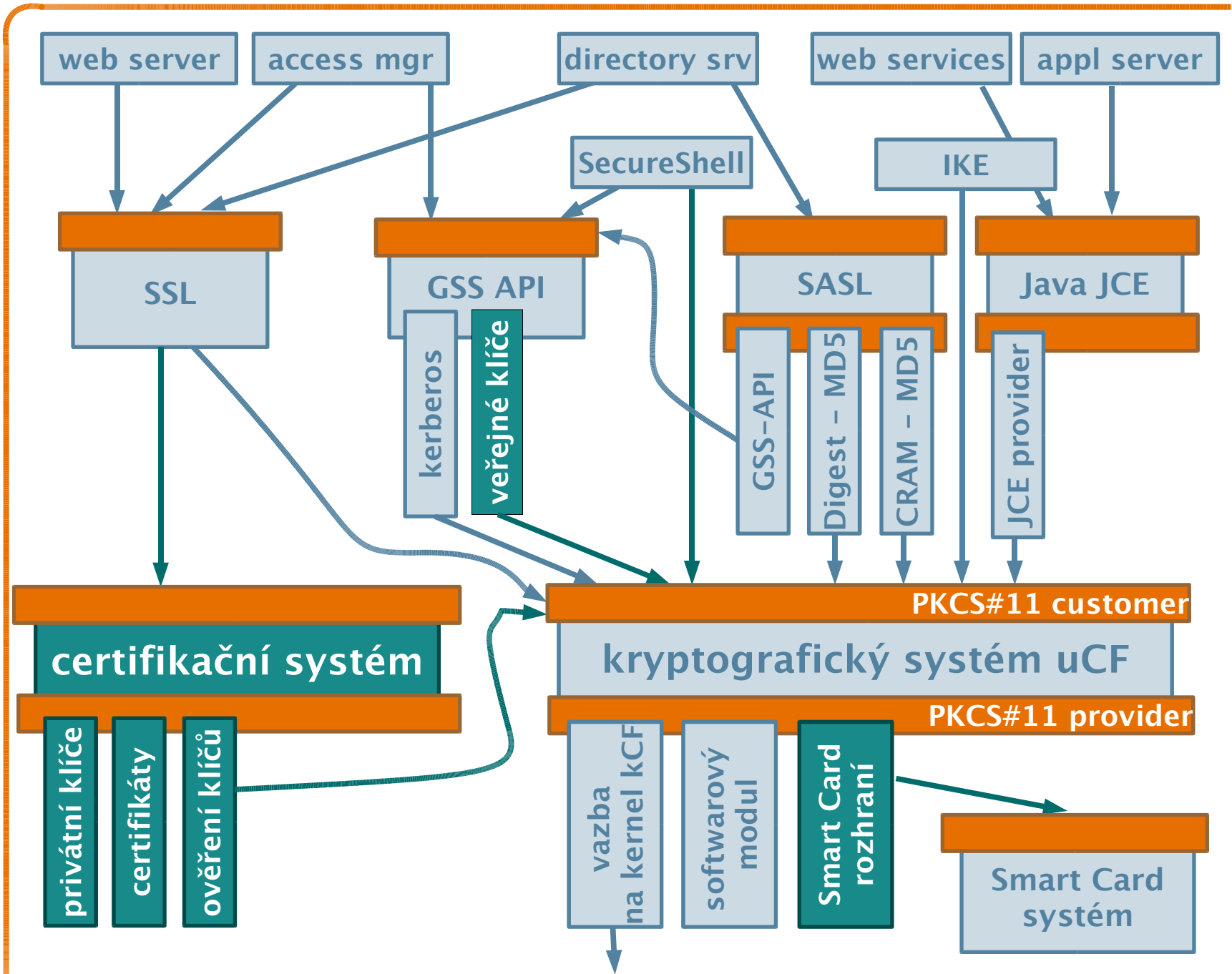
- důvody
  - sjednocení kryptografických knihoven
  - jednotné API (PKCS#11v2)
  - škálovatelnost a vyšší stabilita
  - propojení s HW akcelerátory
  - možnost centrální administrace
- architektura
  - infrastruktura v uživatelské vrstvě (uCF)
  - infrastruktura v kernelu (kCF)

opensolaris <user group>

uCF

krypto

architektura

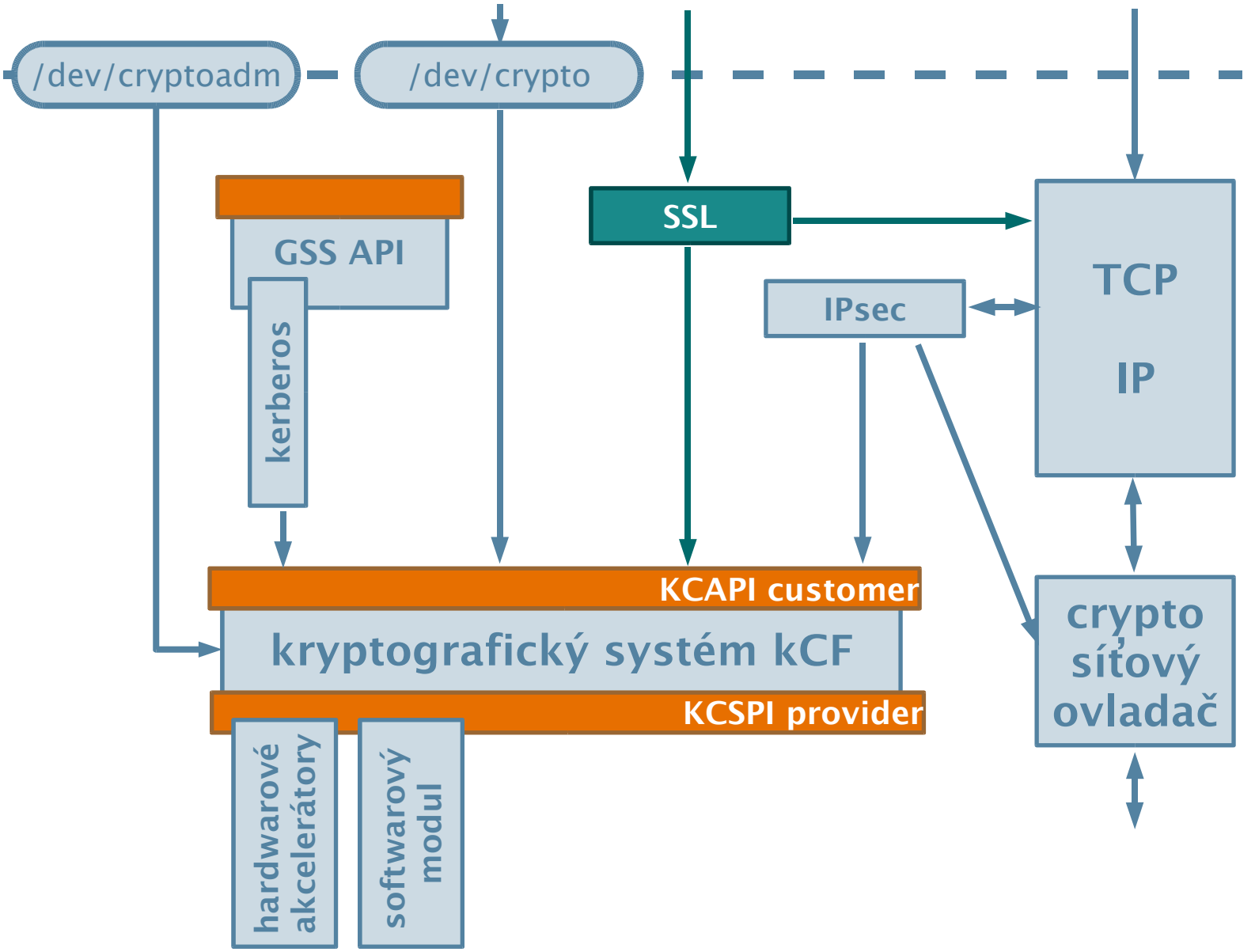


opensolaris <user group>

kCF

krypto

architektura



budoucí rozšíření

opensolaris <user group>

Možnosti

krypto

- konfigurace
  - cryptoadm(1m)
  - například výběr FIPS140 algoritmů
- uživatelské nástroje
  - encrypt(1), digest(1), mac(1) a pktool(1)
- algoritmy
  - pkcs11_softtoken.so
    - DES, 3DES, AES, RC4 (<=128 bit)
    - RSA, DSA, DH
    - MD5, SHA1, SSL HMAC
  - pkcs11_softtoken_extra.so
    - >128 bit

opensolaris <user group>

krypto

DEMO >>>>

opensolaris™ <user group>

dotazy

díky za pozornost

innovate on

opensolaris™