Kryptograficzne zabezpieczanie danych dyskowych we FreeBSD na przykładzie klas GSHSEC i GELI

Paweł Jakub Dawidek
<pjd@FreeBSD.org>
FreeBSD committer

MeetBSD 2005
Kraków

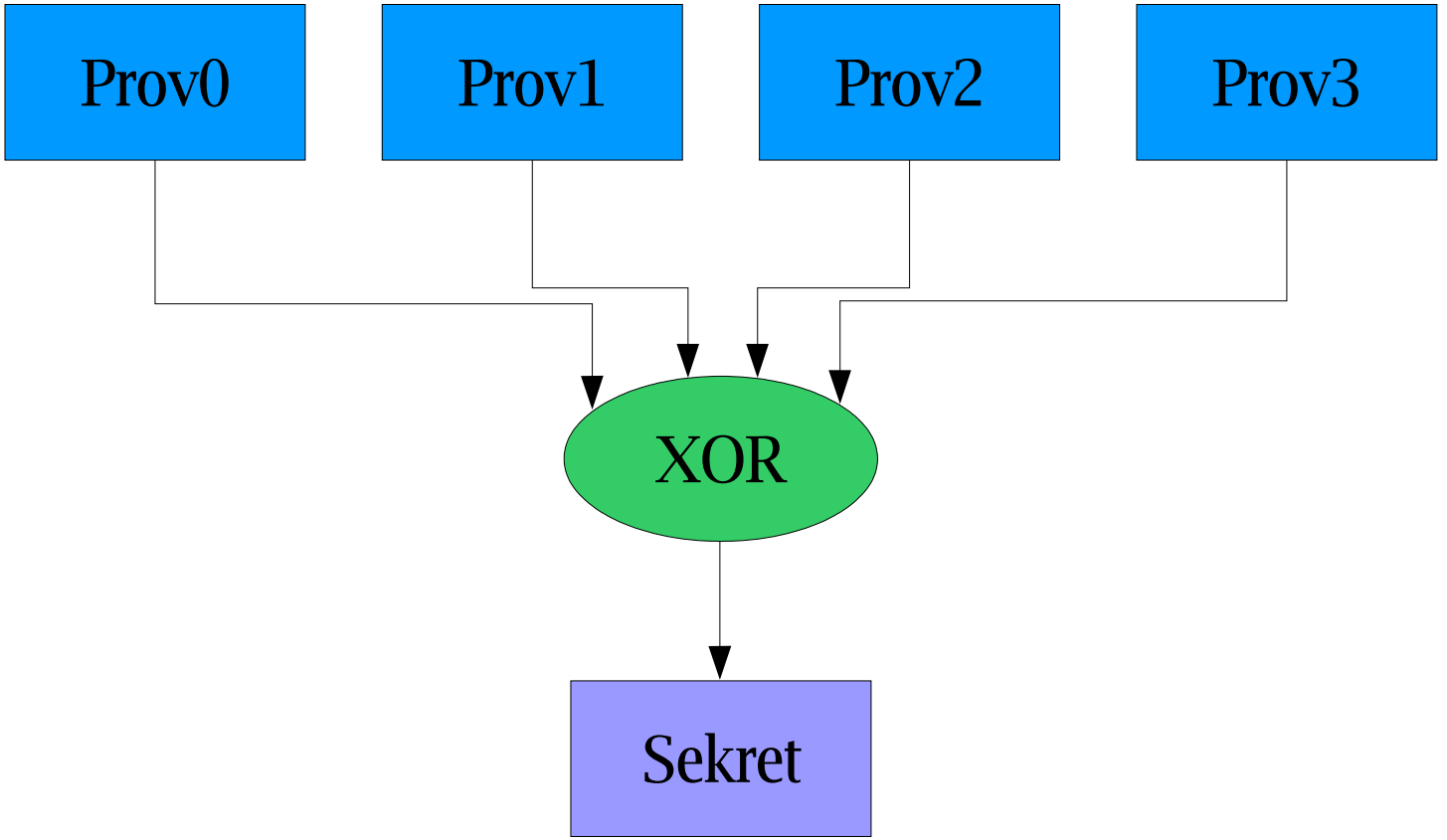
The Power To Serve!

WHEEL open technologies

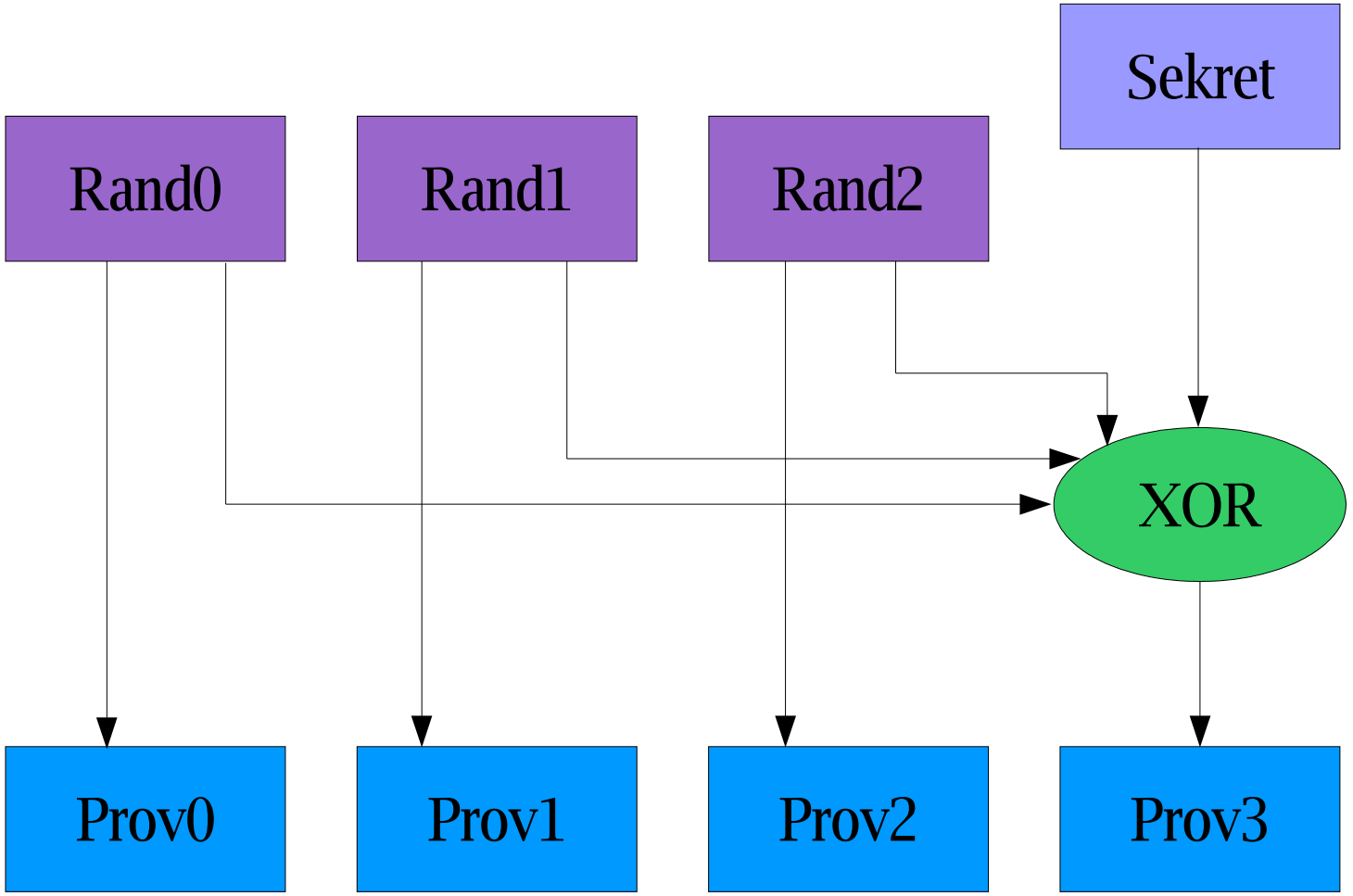
GSHSEC: Podział sekretu

- dzieli sekret na dowolną ilość części
- bez chociaż jednej części nie można odzyskać nawet fragmentu sekretu
- jeśli wszystkie komponenty sekretu są obecne, automatycznie tworzony jest prowajder zawierający sekret, np. /dev/shsec/private

GSHSEC: Działanie (odczyt)



GSHSEC: Działanie (zapis)



GSHSEC: Konfiguracja

```
# gshsec label private ad0s1e da0
# newfs /dev/shsec/private
```

GSHSEC: Wymagania

- bardzo duże zapotrzebowanie na entropię
- rozmiar sekretu = rozmiar jednego komponentu
- w miarę bezawaryjne środowisko

GELI: Klasa GEOM-owa

- może chronić dowolnego prowajdera
- tworzy nowego prowajdera zgodnie ze schematem: <nazwa>.eli, np. da0.eli, mirror/data.eli
- redukuje rozmiar źródłowego prowajdera o jeden sektor
- niezależny od/niewidoczny dla file systemu
- chronii “zimne dyski”

GELI: Używane algorytmy kryptograficzne

- AES, Blowfish, 3DES (w trybie CBC)
- HMAC/SHA-512
- PKCS#5v2/SHA-1

GELI: Wsparcie sprzętowe

- wykorzystuje crypto(9)
- podłączony akcelerator kryptograficzny jest wykrywany i używany automatycznie
- brak akceleratora – dedykowany wątek w kernelu wykonujący operacje kryptograficzne

GELI: Klucz zabezpieczający

- dwa niezależne klucze (użytkownika i firmowy?)
- mogą być zmieniane bez potrzeby przeszyfrowywania zawartości dysku
- może pochodzić z różnych źródeł, np. hasło użytkownika i/lub losowe dane z pliku (“coś co wiesz + coś co masz”)

GELI: Najsłabsze ogniwo – hasło użytkownika

- domyślnie zabezpieczane przy użyciu PKCS#5v2/SHA-1
- 262144 powtórzenia = 2^18 = 1 sekunda

GELI: Szyfrowana partycja /

- katalog /boot/ na USB Pen-Drive lub CD-ROMie
- pytanie o hasło przy boocie

GELI: Klucze jednorazowe

Zastosowanie:

- swap
- file systemy tymczasowe

GELI: Inne opcje

- automatyczne kasowanie kluczy i usuwanie odszyfrowanego prowajdera w momencie ostatniego zamknięcia
- możliwość backupu i odzyskiwania głównych kluczy
- możliwość szybkiego kasowania głównych kluczy

GELI: Wydajność

- proste szyfrowanie sektor w sektor – brak narzutu na dodatkowe/zmodyfikowane operacje I/O
- możliwość użycia wspomagania sprzętowego

GELI: Konfiguracja (1)

```
# dd if=/dev/random of=/mnt/pendrive/da2.key bs=1k count=1
# geli init -K /mnt/pendrive/da2.key /dev/da2
Enter new passphrase:
Reenter new passphrase:
# geli attach -k /mnt/pendrive/da2.key /dev/da2
# dd if=/dev/random of=/dev/da2.eli bs=1m
# newfs /dev/da2.eli
# mount /dev/da2.eli /mnt/secret
...
# umount /mnt/secret
# geli detach da2.eli
```

GELI: Konfiguracja (2)

```
# geli init /dev/da2
Enter new passphrase:       (Twoje hasło)
Reenter new passphrase:
# geli setkey -n 1 /dev/da2
Enter passphrase:           (Twoje hasło)
Enter new passphrase:       (Hasło Twojej dziewczyny)
Reenter new passphrase:
...
# geli detach da2.eli
```

GELI: Konfiguracja (3)

```
# dd if=/dev/random of=/dev/ad0s1b bs=1m
# geli onetime -d -a 3des ad0s1b
# swapon /dev/ad0s1b.eli
```

Status i dostępność

- gshsec(8) dostępne od FreeBSD 5.4
- geli(8): http://perforce.FreeBSD.org

Koniec

Pytania?