CryptoPanel

edycja #8

Porządkowanie rozproszonych kluczy szyfrujących wewnątrz organizacji i zarządzanie nimi z poziomu jednego repozytorium.

CryptoPanel

dziś dyskutują

Piotr Majek

Security Specialist
piotr.majek@clico.pl
mob. +48 663 994 996

Artur Holeczek

Partner Account Manager /
Product Manager
artur.holeczek@clico.pl
mob. +48 667 699 444

CryptoPanel

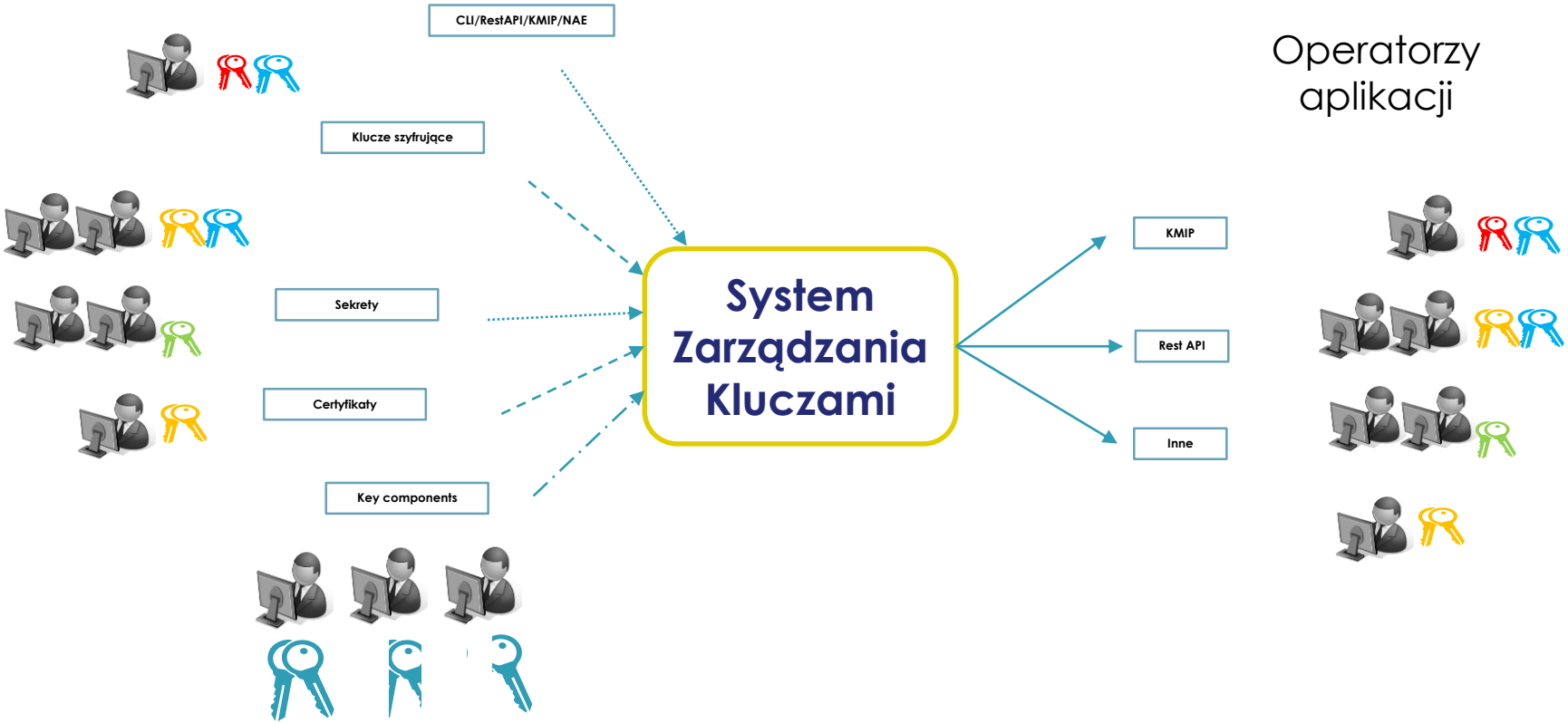
Dzisiejszy problem

Co nas boli...

- Jesteśmy instytucją rynku finansowego;
- Z racji swojej działalności posługujemy się ogromną ilością kluczy szyfrujących w rozwiązaniach IT i paymentowych, generowanych często z wykorzystaniem darmowych narzędzi programowych;
- Klucze szyfrujące wykorzystywane są w wielu różnych systemach i aplikacjach oraz często brakuje im wspólnego interfejsu (np. KMIP czy RestAPI);
- Klucze zarządzane i wykorzystywane są przez określone zespoły ludzkie (musimy utrzymać ten podział!)

- ...poszukujemy rozwiązania do realizacji centralnego systemu zarządzania kluczami.
- ...chcemy zaprzestać przechowywania kluczy w plikach PEM, na papierze (key components). Generalnie: poza centralnym repozytorium.
- ...w pierwszym etapie chcemy stworzyć niezawodne repozytorium kluczy z podziałem uprawnień. Możliwość eksportu i importu materiału kryptograficznego.
- …w kolejnych etapach chcemy wprowadzić pełne zarządzanie cyklem życia kluczy (utworzenie, udostepnienie, monitorowanie, wstrzymanie, odwołanie, archiwizacja, kasowanie)
- …potrzebujemy automatycznej archiwizacji kluczy (najlepiej offsite).
- …chcemy przechowywać także inne „sekrety”, takie jak certyfikaty.

Co posiadamy, a co chcemy posiadać?



CryptoPanel

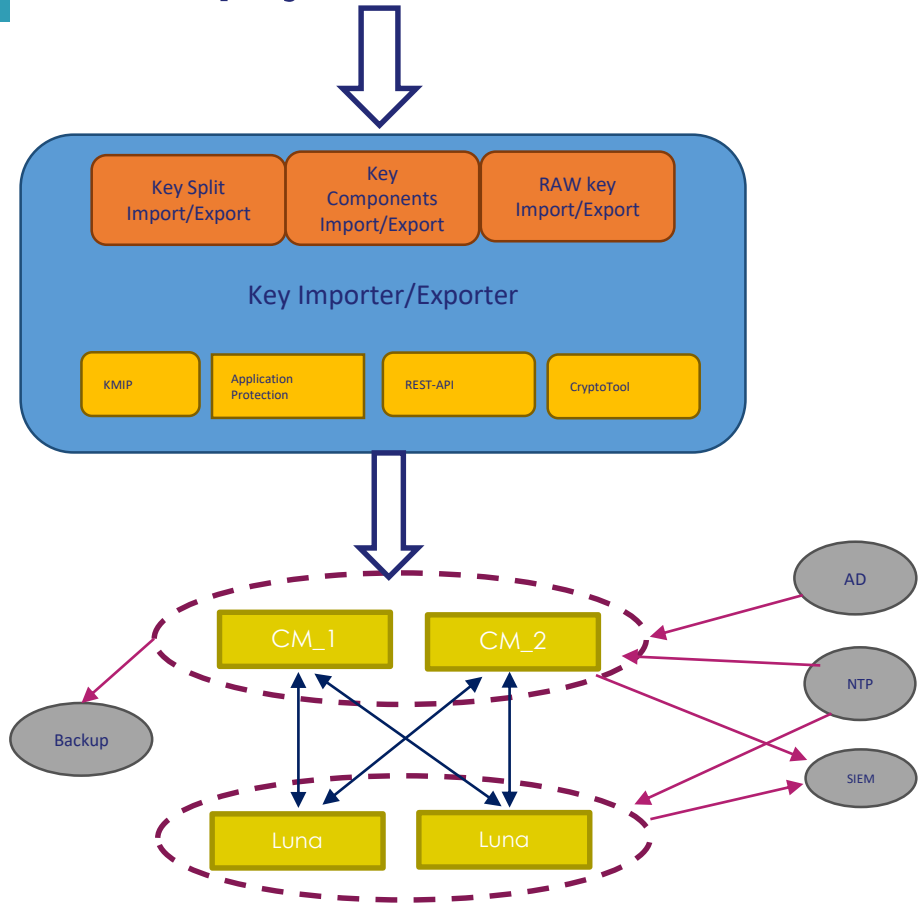
Rozwiązanie

Rozwiązanie:

CipherTrust Manager. I wystarczy!

Dodatkowo – rozbudowa o CipherTrust Application Protection (CAP)

Koncepcja



Koncepcja systemu:

System zaprojektowano jako system 3 warstwowy. Sprzętowy HSM (Luna) zapewnia źródło zaufania do generowania/przechowywania kluczy. CipherTrust Manager zapewnia większość funkcji zarządzania kluczami (dostępne za pośrednictwem REST-API i CryptoTool). Jednak niektóre możliwości importu/eksportu kluczy muszą zostać zaprojektowane i zakodowane (np. podział kluczy, kluczowe komponenty itp.)

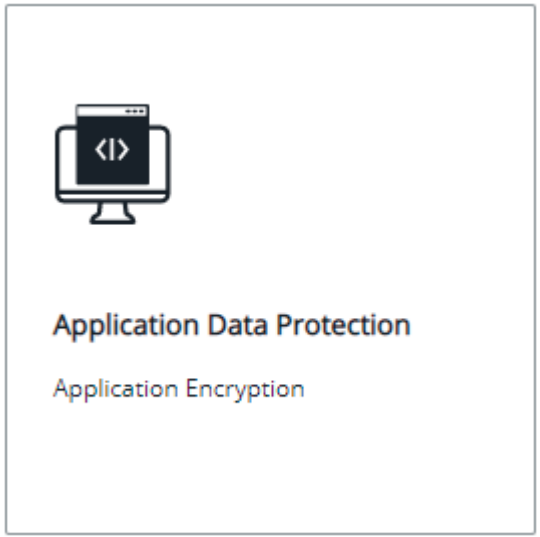
1. Key Import/Export (KIE) – komponent należy zaprojektować i zakodować. Wykorzystuje Application Protection do niektórych podstawowych funkcjonalności od razu po wyjęciu z pudełka (np. eksport/import klucza RAW), inne muszą zostać zaprojektowane i zakodowane.
2. CipherTrust Manager (klaster) – dostarczany jako urządzenie wirtualne zapewnia najwięcej możliwości zarządzania kluczami (generowanie, użycie, zniszczenie).
3. Luna (klaster) – sprzętowy HSM zapewniający źródło zaufania do generowania i ochrony kluczy

THALES

Co może CipherTrust Manager w kwestii importu/eksportu kluczy?

Możliwości CipherTrust Managera

- „Z pudełka” mamy dostęp do obsługi zapytań REST-API,
- Możliwość rozbudowy o konektor obsługujące inne biblioteki – CipherTrust Application Protection.



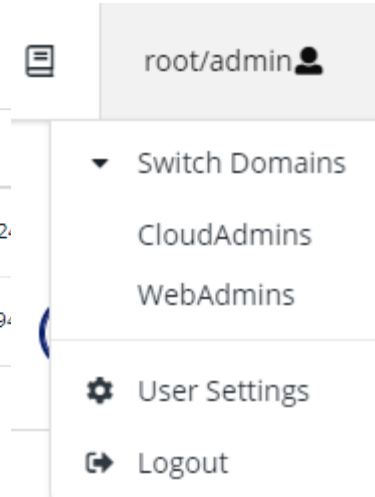
CipherTrust Manager i domeny - czyli przydział kluczy do konkretnych grup użytkowników

Domeny – czym są i co nam dają?

- Domena – możliwość separacji kluczy, użytkowników i aplikacji w ramach jednego CipherTrust Managera,
- Zwiększona elastyczność zarządzania,
- Osobne domeny – brak „mieszania się” kluczy, niezależne kopie zapasowe.

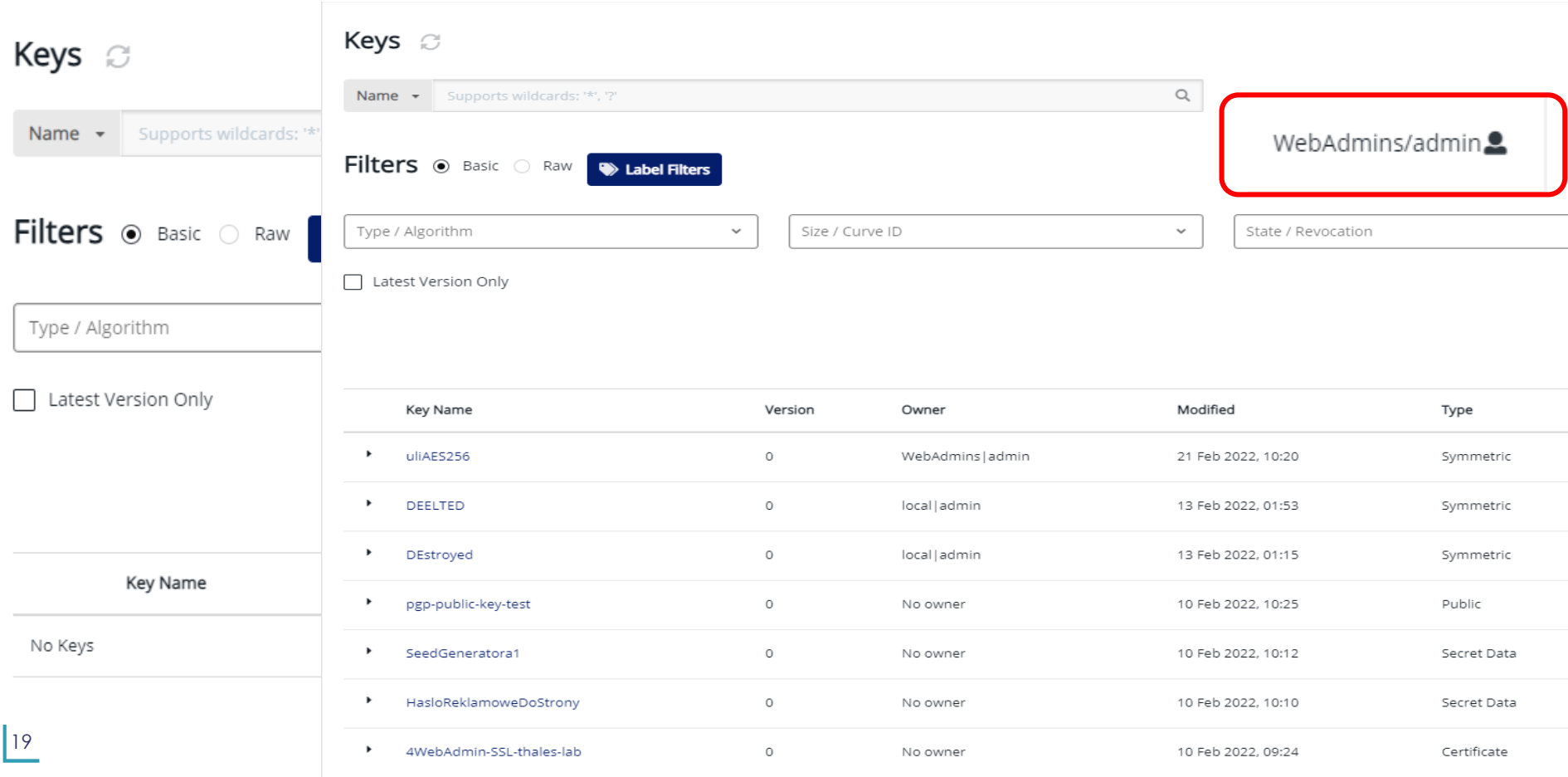
Domeny – zmiana zasobnika kluczy „w locie”

Domains



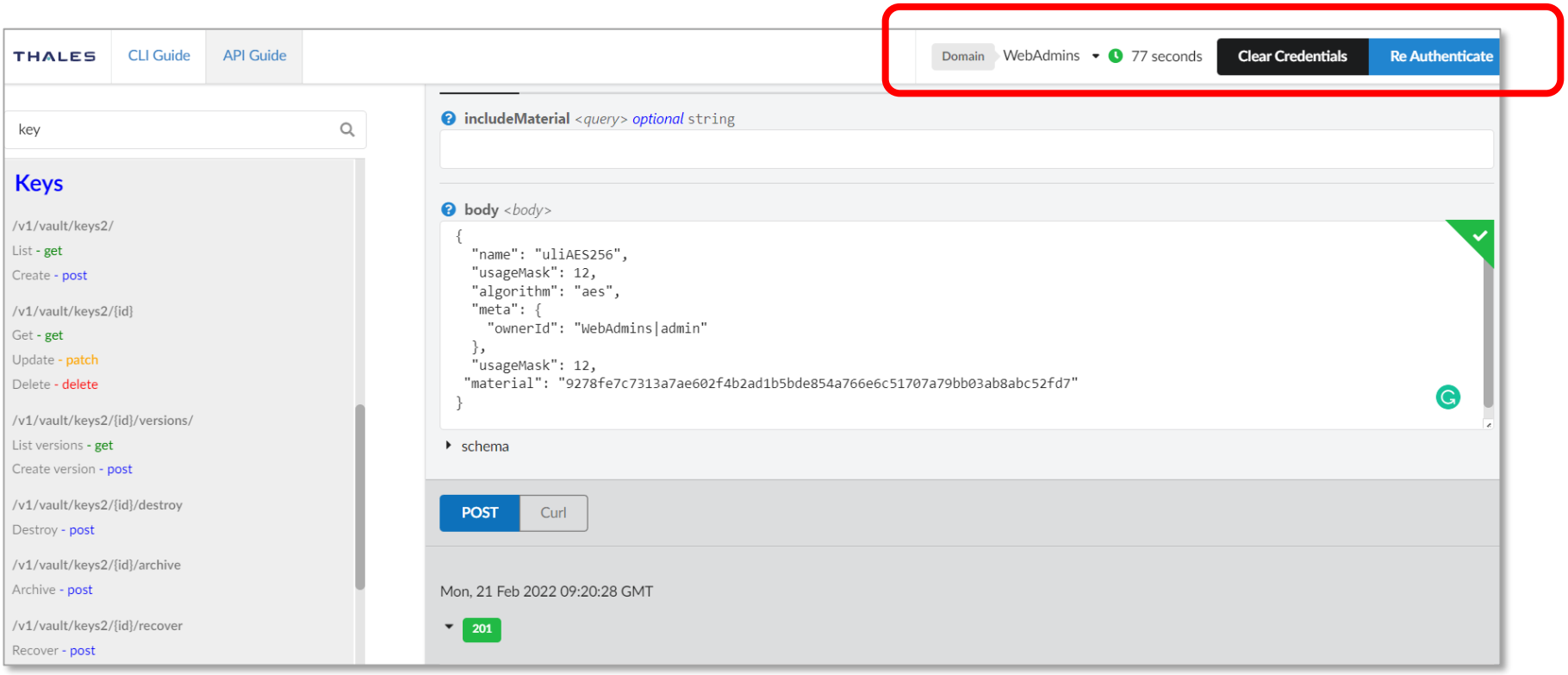
Name	Id	Parent Domain	CA
CloudAdmins	74fdaf0c	root	98b0992
WebAdmins	3eeacccb	root	173ea69

Domeny – podział kluczy

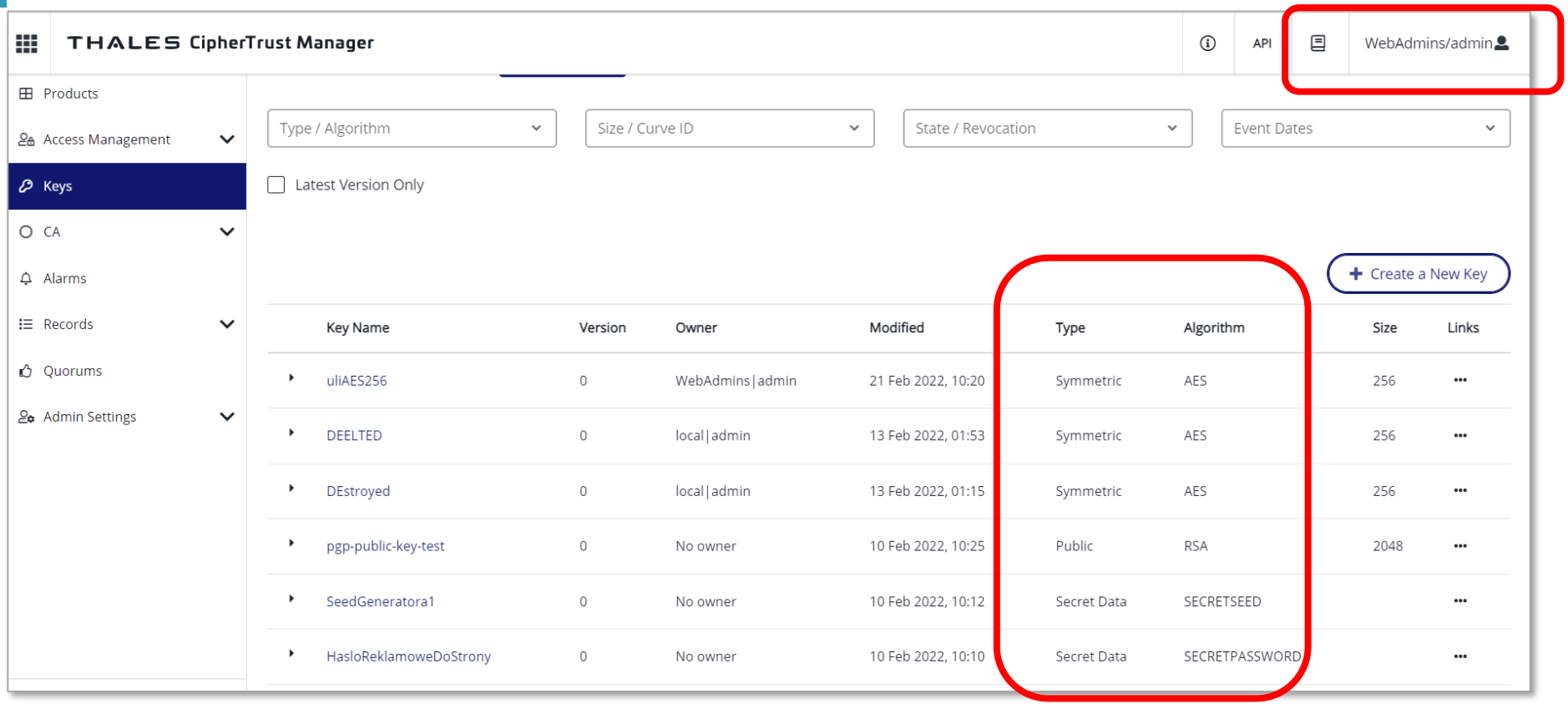


Jak załadować różne klucze do CipherTrust Managera:
RestAPI
JCE CryptoTool

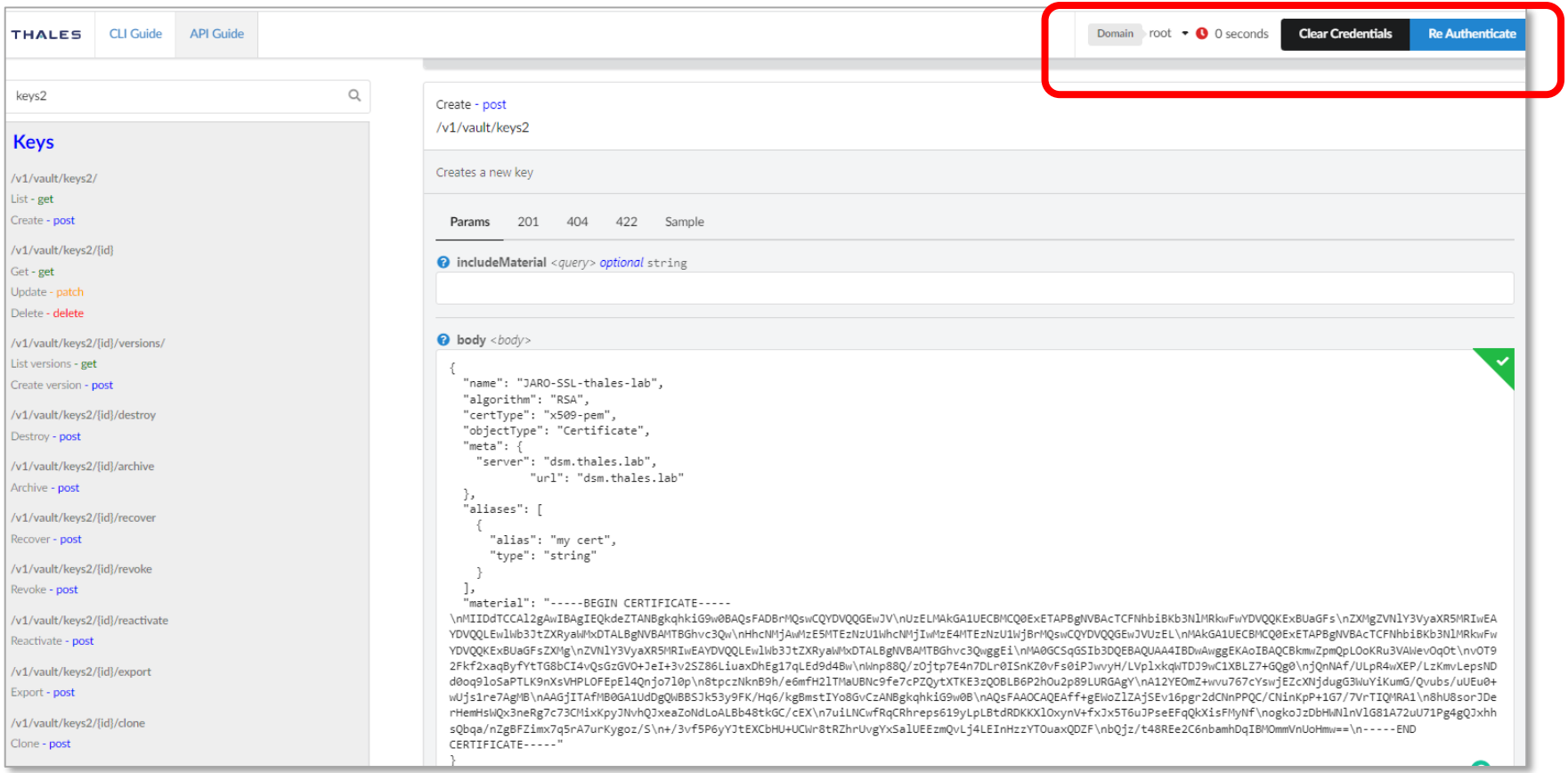
Piaskownica - przez RestAPI, załadujesz klucze i sekrety…



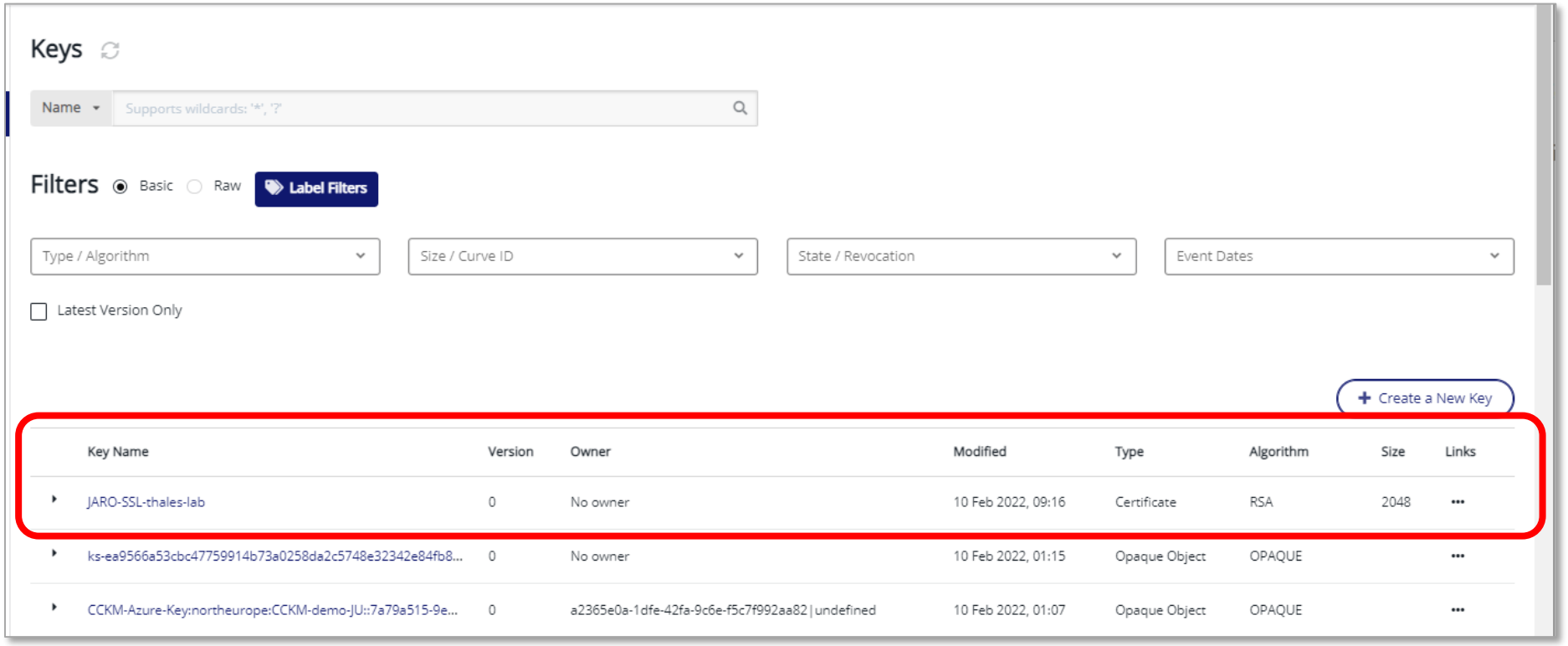
Piaskownica - przez RestAPI, załadujesz klucze i sekrety



Piaskownica – przez RestAPI, załadujesz także certyfikaty!



Piaskownica - przez RestAPI, załadujesz wszystko



Użyj CryptoTool – gdy nie jest niestandardowo…

- CryptoTool może zostać użyty jako narzędzie do debugu, aby zweryfikować poprawną instalację bibliotek SafeNet JCE Provider.

- CryptoTool wykonuje również operacjne kryptograficzne z poziomu linii poleceń lub jako skrypty.

Ważne: KMIP nie jest częścią CryptoTool.

CryptoTool – Opcje

```
C:\Program Files\Java\jre7\bin>java CryptoTool -help

USAGE:
        java CryptoTool OPERATION options

SUPPORTED CRYPTO OPERATIONS:

        ENCRYPT, DECRYPT, MAC, MACV, SIGN, SIGNV

SUPPORTED KEY MANAGEMENT OPERATIONS:

        GENERATE, DELETE, EXPORT, IMPORT, LIST

SUPPORTED OPTIONS:

-in filename
        specify a file instead of stdin
-out filename
        specify a file instead of stdout
-key keyname
        key name
-alg algname
        algorithm
-iv value
        initialization vector when required (must be hex ASCII encoded)
-sig value
        provide signature value as an argument to use for verification (must be
 hex ASCII encoded)
-sigfile filename
        alternative to -sig; provide signature value in a file
-mac value
        provide mac value as an argument to use for verification (must be hex A
SCII encoded)
-macfile filename
        alternative to -mac; provide mac value in a file
-auth username:passwd
        username and password for authentication
-keysize size
        key size to use for key generation
-exportable
        create exportable key
-deletable
        create deletable key
-ip ip
        NAE server IP to use  (can be a colon separated list of IP addresses)
-port port
        NAE server port to use
-protocol prot
        protocol to use (ssl or tcp)
-edgesecure edgesecure
        to specify EdgeSecure name
```

Dostępne opcje w aplikacji dostępne są po dodaniu opcji `-help`.

```
java CryptoTool -help
```


THALES

CryptoTool – wykonywanie przykładowych komend

CryptoTool – Generowanie klucza RSA

```
C:\Program Files\Java\jre7\bin>java CryptoTool GENERATE -auth test1:asdf1234 -ke
y rsakey12 -alg RSA -keysize 2048
Key generated OK
```

```
C:\Program Files\Java\jre7\bin>java CryptoTool GENERATE -auth test1:asdf1234 -ke
y mykey33 -alg AES -keysize 256
Key generated OK
```

CryptoTool – wyświetlenie listy dostępnych kluczy

```
C:\Program Files\Java\jre7\bin>
C:\Program Files\Java\jre7\bin>java CryptoTool LIST -auth test1:asdf1234
CA_Cert : RSA
hmackey : HmacSHA1
hmackey222 : HmacSHA1
key1 : AES
mykey256 : AES
mykey33 : AES
protectDBkey3 : AES
rsakey : RSA
rsakey1 : RSA
rsakey12 : RSA
rsakey2 : RSA
TMHMAC : HmacSHA256
TMKEY : AES
token_hash_key : HmacSHA256
token_key : AES

C:\Program Files\Java\jre7\bin>_
```


THALES

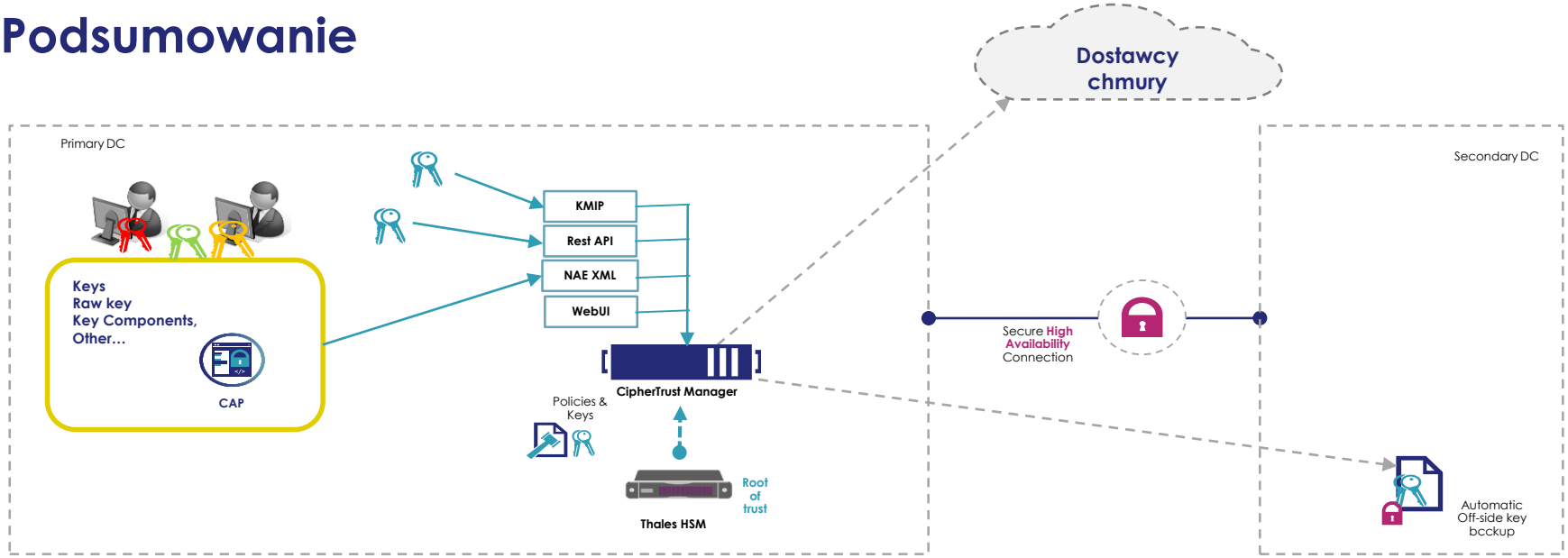
„Nauczki“, czyli *lessons learned i „w dokumentacji nie znajdziecie“*

- Kiedy myślimy o scentralizowanym systemie przechowywania i udostępniania kluczy szyfrujących weźmy **pod uwagę ich dotychczasowy sposób wykorzystania** (np.: papier->KMS->papier – czy warto?)
- Import/Export komponentów: z powodu braku mechanizmu XOR_DATA w CM oraz zabronienia tego mechanizmu (jako niebezpiecznego) w HSM, składanie klucza komponentowego odbywa się w aplikacji w pamięci stacji/systemu. Dlatego należy zadbać aby była to wydzielona stacja administracyjna objęta dodatkowymi zabezpieczeniami.
- Czy wykorzystanie RestAPI wymaga licencji? Nie! ☺

CryptoPanel

podsumowanie

Podsumowanie



- Wystarczy licencja na CM
- Dostępny model subskrypcyjny lub dożywotni
- Licencja demo do testów dostarczana wraz z produktem
- Nie trzeba ale warto: HSM

- CipherTrust Manager – 14,4k Euro netto
- CipherTrust Flex Connector - Advanced, Perpetual – 8,6 k Euro netto

Materiały

- Oprogramowanie do pobrania lub wersja ewaluacyjna...
  - Bardzo prosimy o kontakt z nami!
- Dokumentacja...
  - CipherTrust Manger: https://www.thalesdocs.com/ctp/cm/latest/
- Zapraszamy także na... Kolejny CryptoPanel
- QUIZ Z NAGRODAMI
  https://forms.office.com/r/sK0RYRzHi1