PKI

A public key infrastructure (PKI) is a system for the creation, storage, and distribution of digital certificates which are used to verify that a particular public key belongs to a certain entity. The PKI creates digital certificates which map public keys to entities, securely stores these certificates in a central repository and revokes them if needed. (Wikipedia)

Why

To allow organizations to be independent in keys management with proper cyber security levels of protection and controlled costs in the years.

What

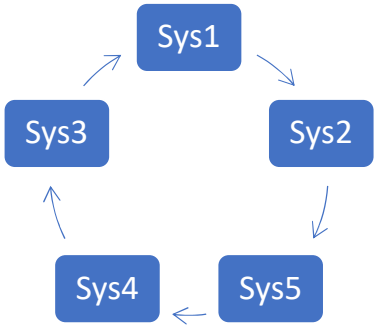
W1 PKI product and solution scale from low-end systems up to full cloud farms.

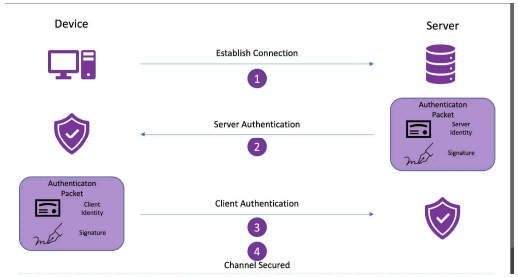
How

W1 PKI supports all flavors of deploy:

- In customer premises
- In customer cloud
- PAAS
- SAAS

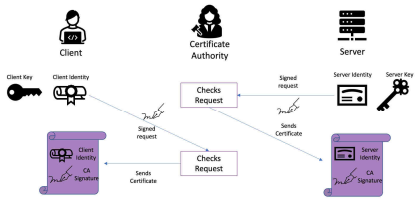
Protect communication among systems / implement zero trust approach

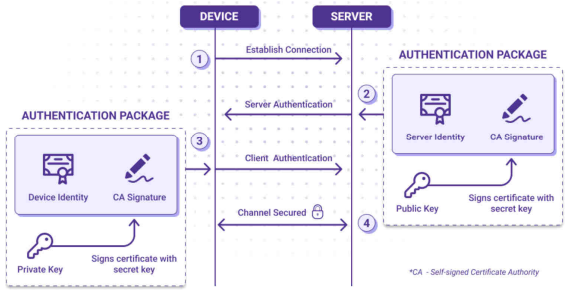




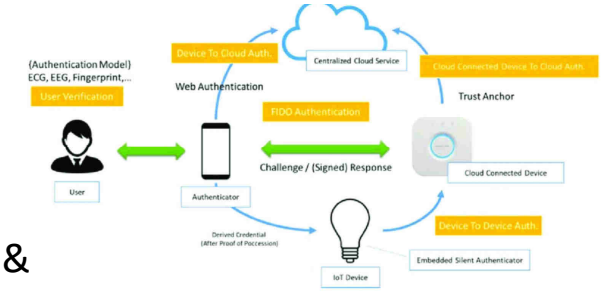
Ensure authenticity, integrity and confidentiality in client-server communications

Digital sign and validate data and documents.

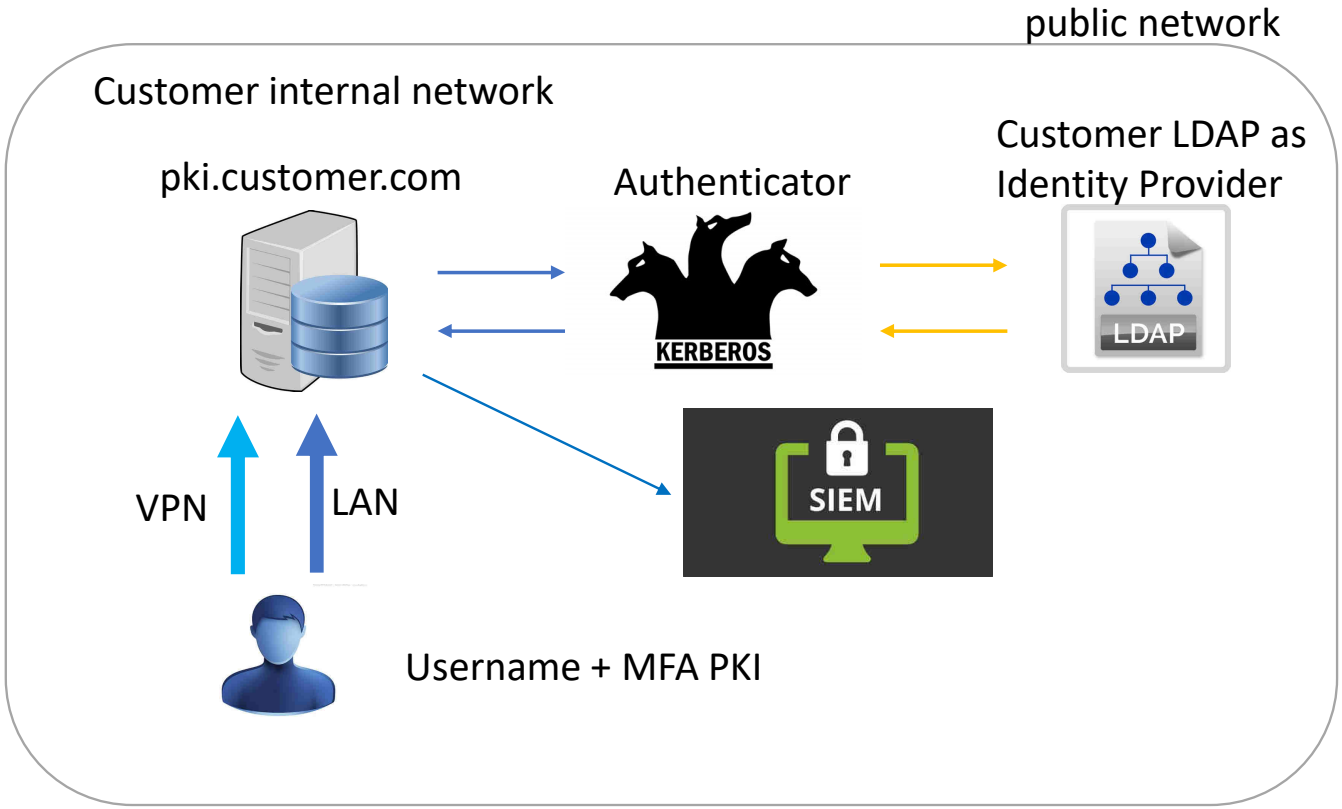




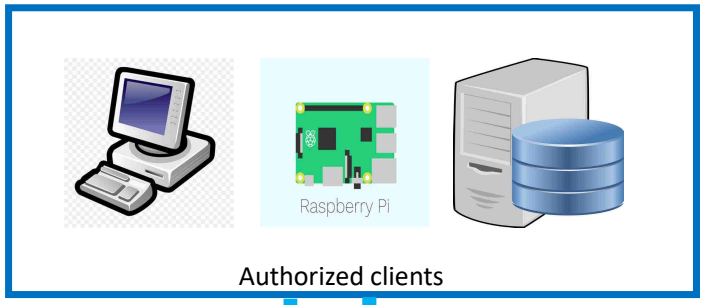
Implement repudiation and key-rolling automatic management and distribution



Enroll digital identities on mobile apps & devices, IoT devices



HIGH LEVEL PROPERTIES
Integrated with customer identity and security provider
Reachibility according to customer design
Multiplatform capability
Siem feeder
Physical installation or contained installation as well



VPN

LAN

pki.while1.com

VPN

LAN

CSR signature
Automated build of chains
CSR multiple submission and public certificate download
Choice of chain

Submit CSR

Api
Get expiring certificates
Get CRL files

Generic API

Creation of leafs and/or identities

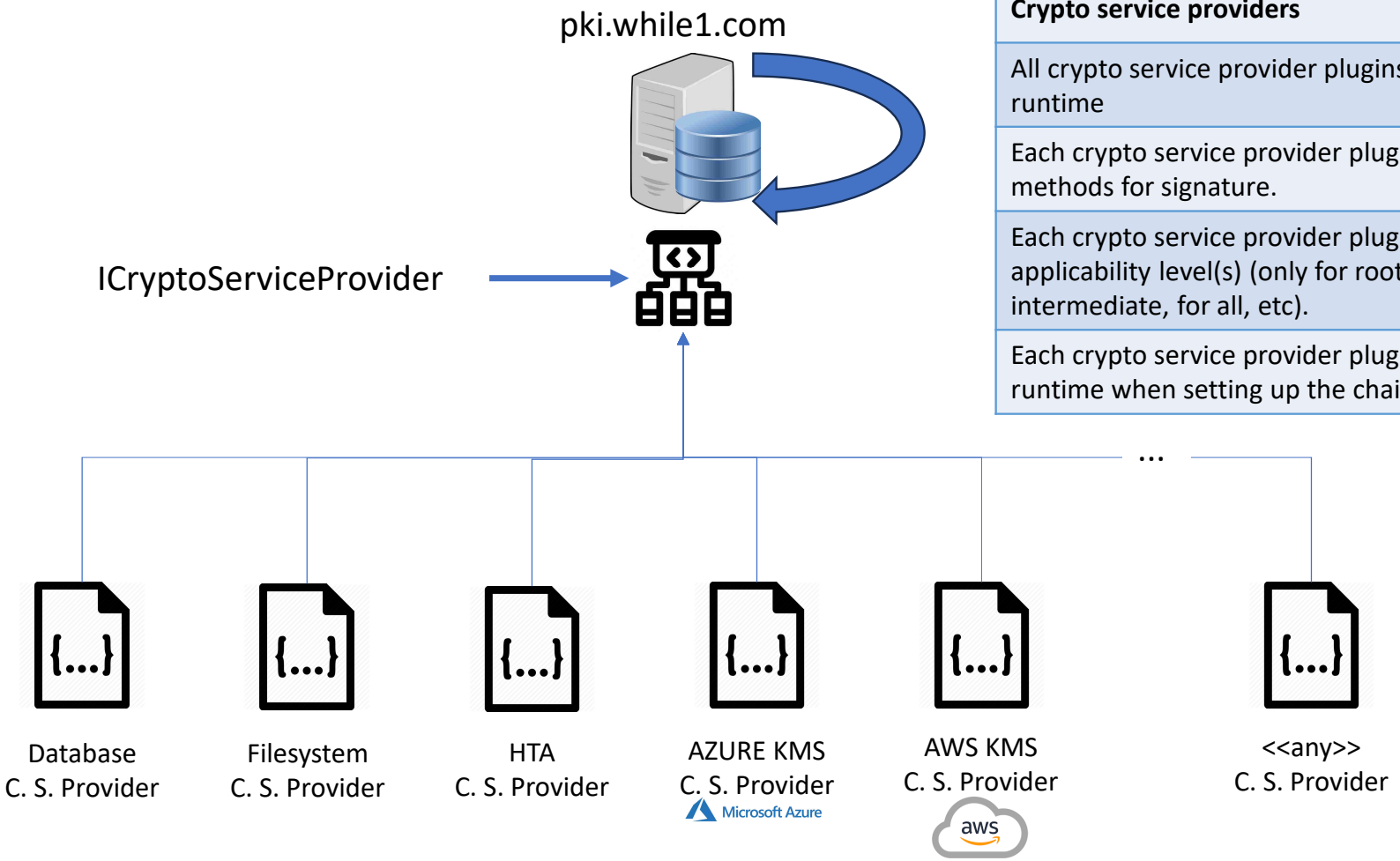
Certificates
Choice of signature algorithm
Choice of intermediate
Choice of crypto provider
Choice of duration (less than intermediate)
Each certificate can be revoked

Creation of intermediate CA

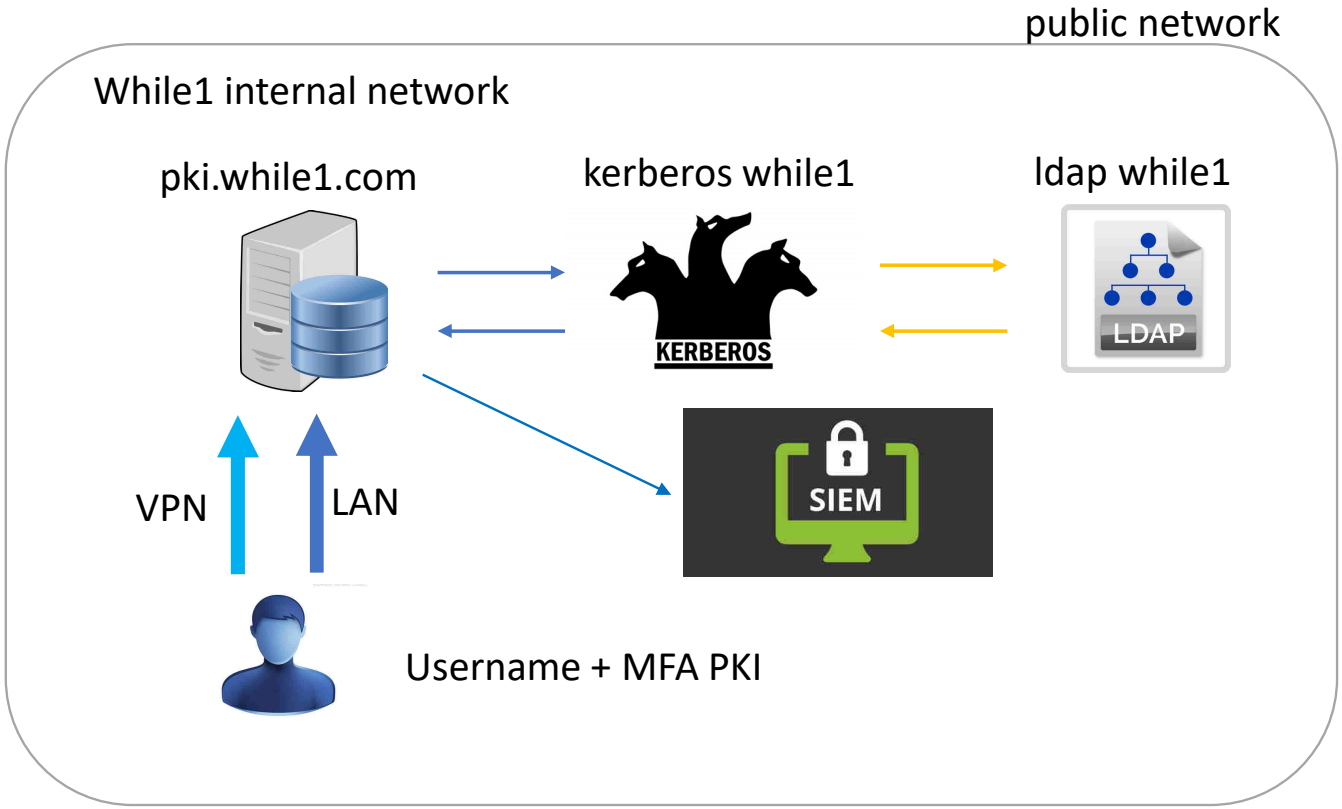
Intermediate (s)
Choice of signature algorithm
Choice of root ca
Choice of crypto provider
Choice of duration (< root)
Multiple levels of creation and revocation.

Creation of root CA

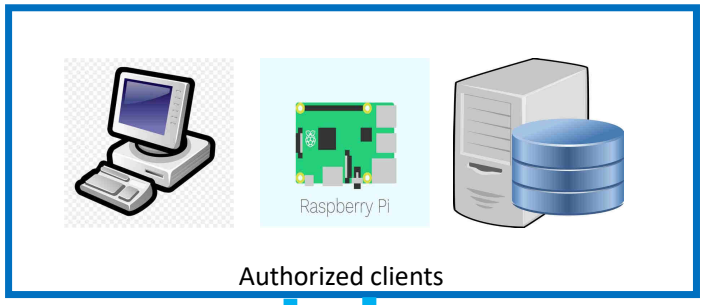
root ca
Choice of signature algorithm
Choice of crypto provider
Choice of duration
Revocation will revoce sub levels



Crypto service providers
All crypto service provider plugins will be loaded runtime
Each crypto service provider plugin will define methods for signature.
Each crypto service provider plugin will define its applicability level(s) (only for root, for root and intermediate, for all, etc).
Each crypto service provider plugin will be usable at runtime when setting up the chain.



HIGH LEVEL PROPERTIES
Integrated with internal security provider
Not exposed to internet
Multiplatform capability
Siem feeder
Physical installation or contained installation as well



VPN

LAN

pki.while1.com

VPN

LAN

CSR signature
Automated build of chains
CSR multiple submission and public certificate download
Choice of chain

Submit CSR

Api
Get expiring certificates
Get CRL files

Generic API

Creation of leafs and/or identities

Certificates
Choice of signature algorithm
Choice of intermediate
Choice of crypto provider
Choice of duration (less than intermediate)
Each certificate can be revoked

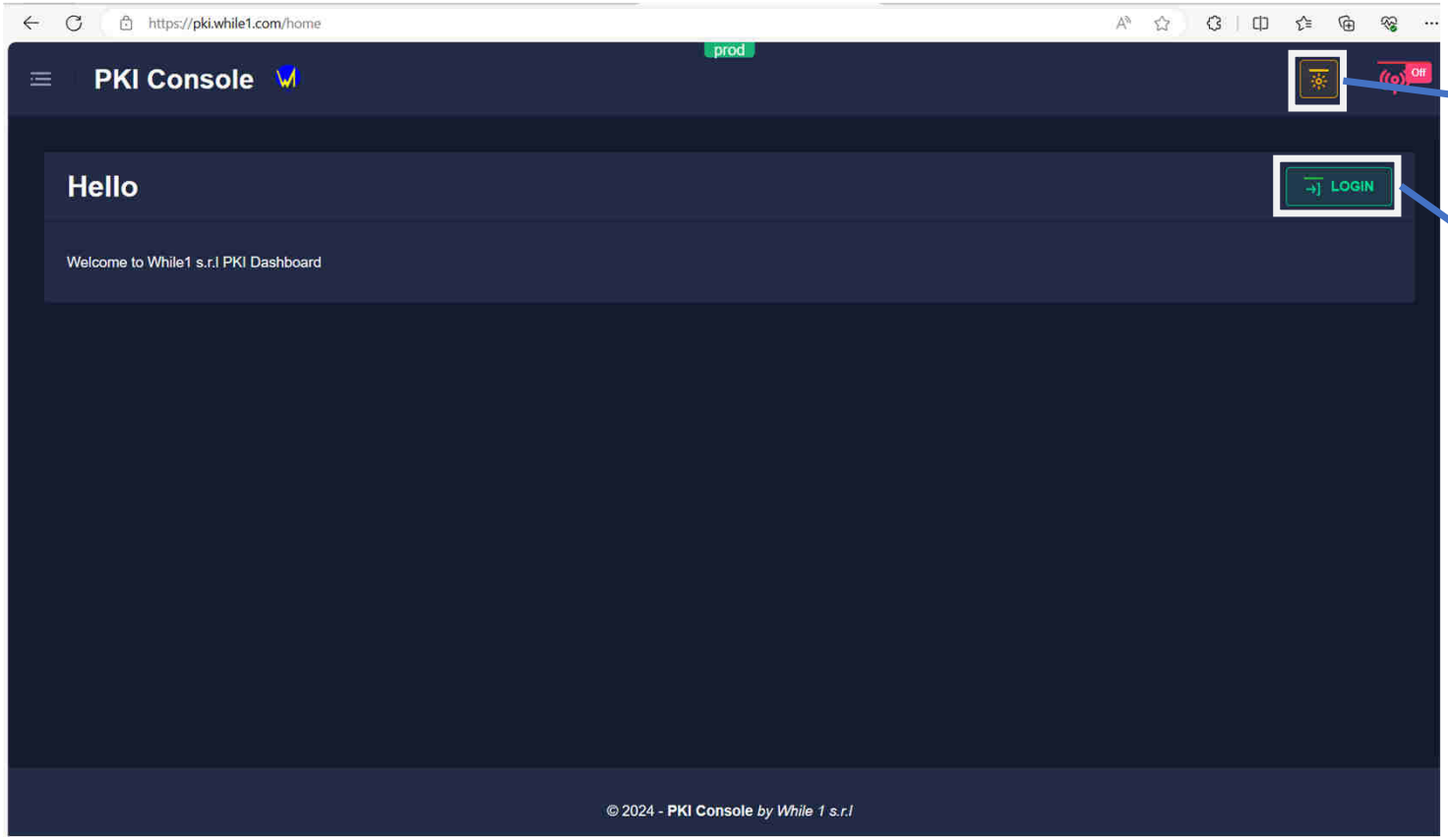
Creation of intermediate CA

Intermediate (s)
Choice of signature algorithm
Choice of root ca
Choice of crypto provider
Choice of duration (< root)
Multiple levels of creation and revocation.

Creation of root CA

root ca
Choice of signature algorithm
Choice of crypto provider
Choice of duration
Revocation will revoce sub levels

while1.com // PKI >> Welcome page



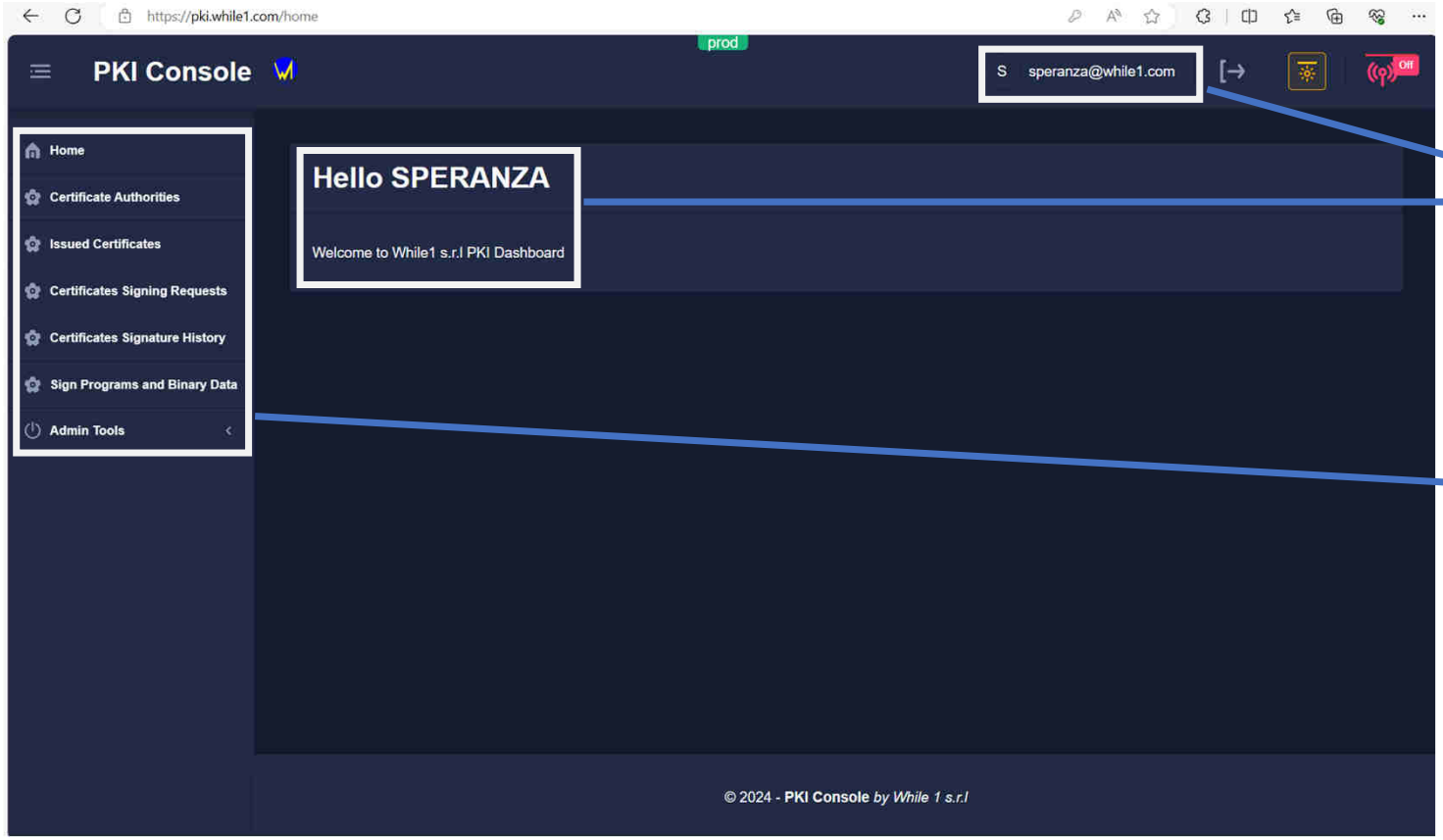
White or black layout

Proceed to login page.
Login providers supported are:

- ADFS
- LOCAL Provider
- Kerberos (over LDAP)

Other providers can be added with dedicated development

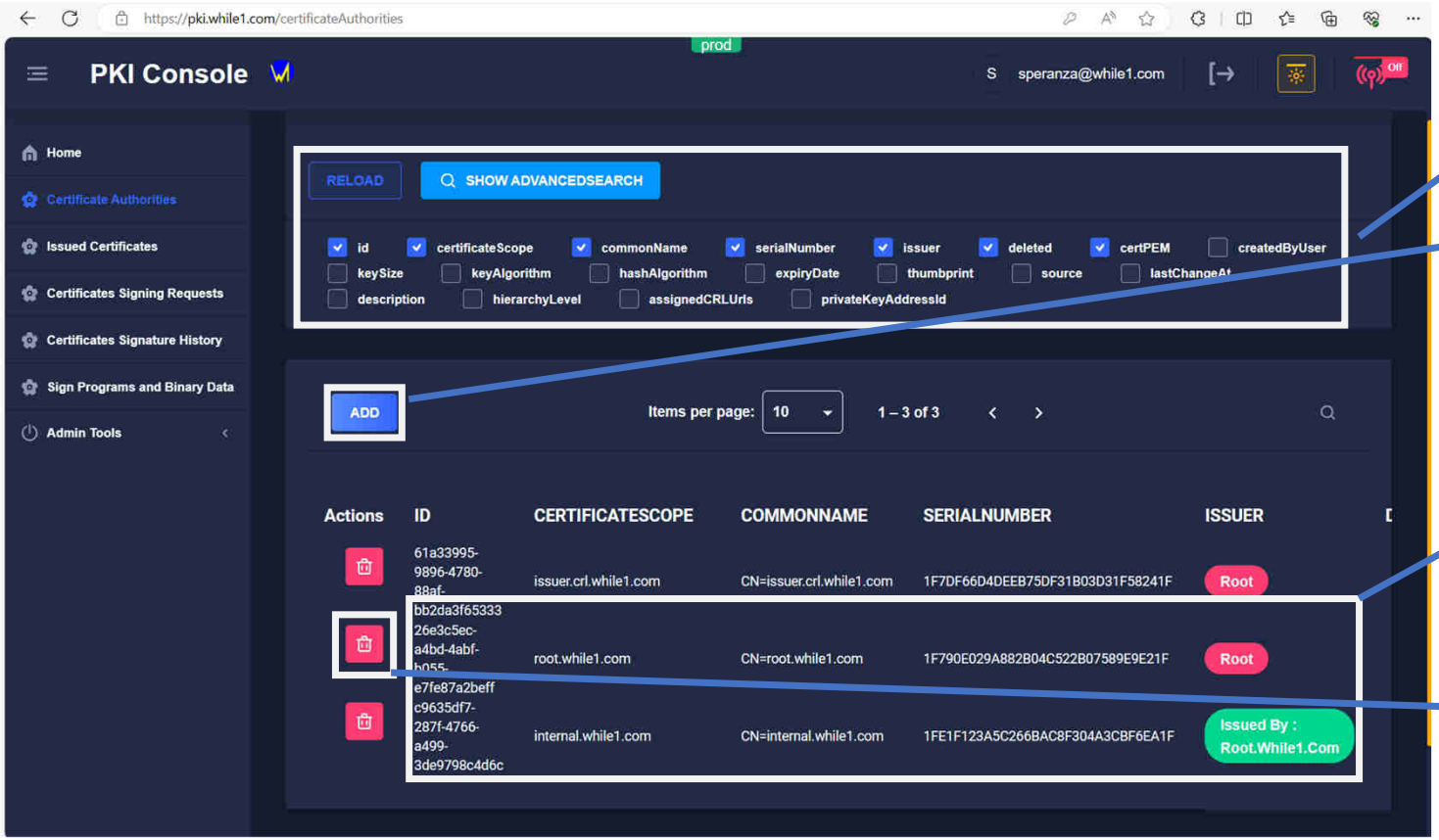
while1.com // PKI >> Activities



Reference to logged user

List of actions that user can perform. Authorization for user is bases on functionality

while1.com // PKI >> root and intermediates



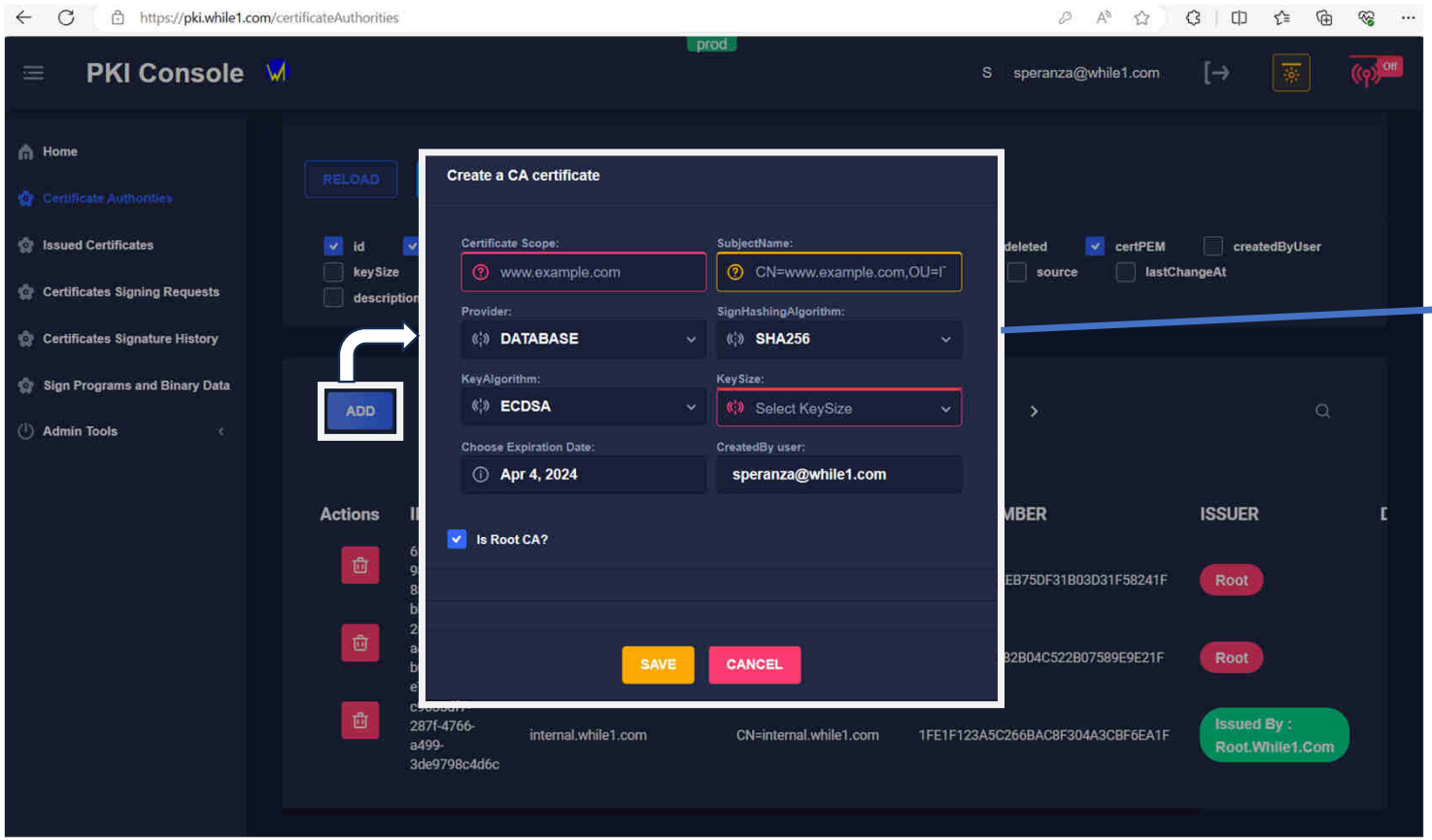
Filter for search. All fields are filterable and produce immediate result

Button to add root or intermediate. Explain in the next slide

Minimum details for each issued root or intermediate certificate. If the certificate is a root certificate, the issuer will itself, otherwise the name of the issuer (for example an intermediate)

The delete button will delete the certificate. If level has sub levels, you can delete all signed certificate. The delete will automatically populate the CRL file.

while1.com // PKI >> Add new certificate

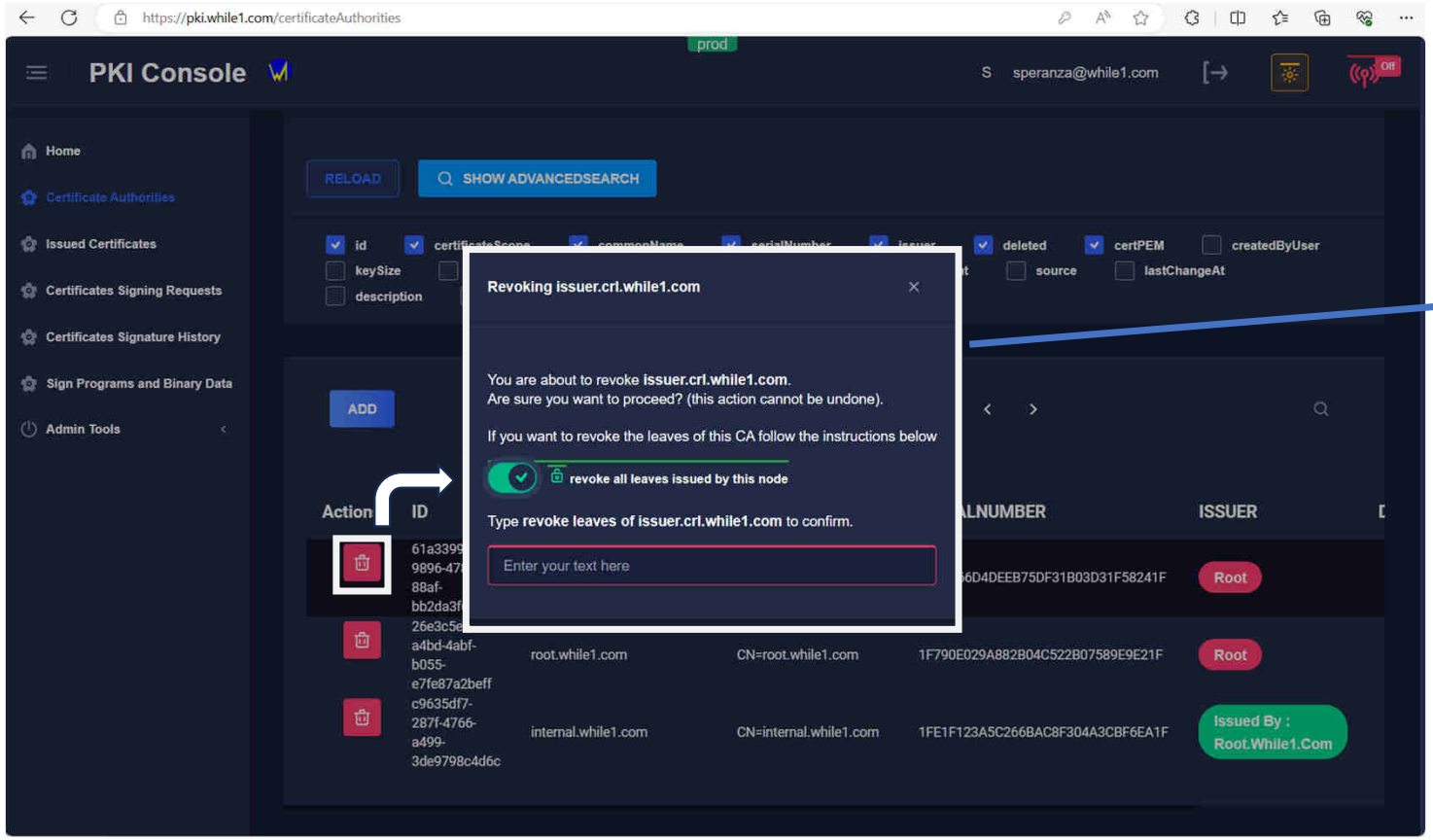


In this dialog we add a root certificate or a intermediate certificate. Properties are:

- Certificate scope → a label (root.while1.com)
- Subject name → if empty, automatically built, else it can be fully specified
- Provider → a list of possible signature provider*
- SignHashingAlgorithm, KeyAlgorith, Keysize → possible combination of valid values and size
- Choose Expiration Date → you can choose the expiration date that best fit the needs.
- Is Root CA → if selected, the certificate is a root certificate, if not selected, the certificate is intermediate and a root can be selected.

* Providers can be developped to fit a need. An interface is shared. The implementation of that interface will allow to have a new available provider

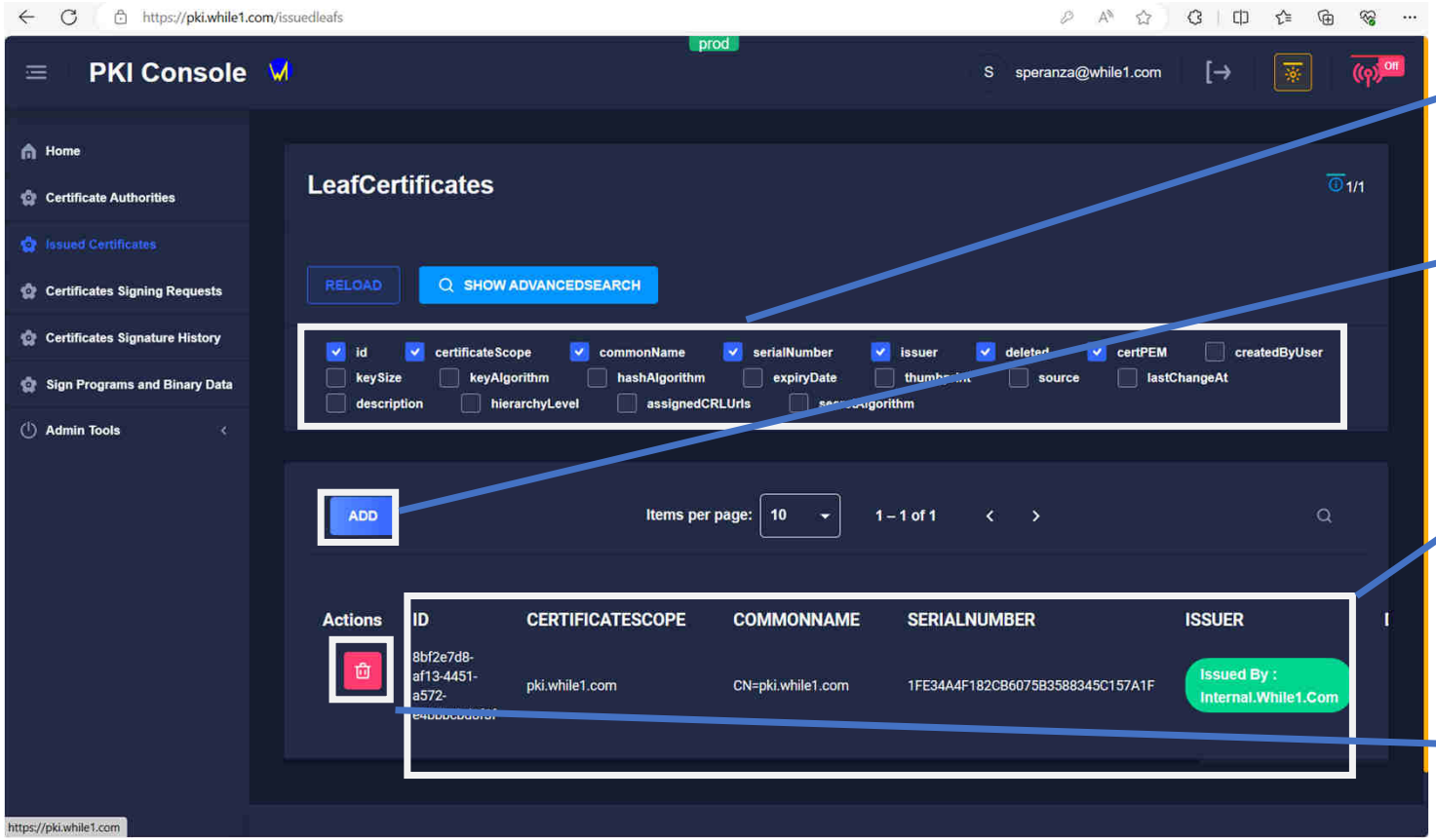
while1.com // PKI >> delete a certificate



The deletion of a certificate requires a strong confirmation because:

- All signed certificates will be deleted
- All deleted certificates will be added to CRL file

while1.com // PKI >> manage leaf certificates



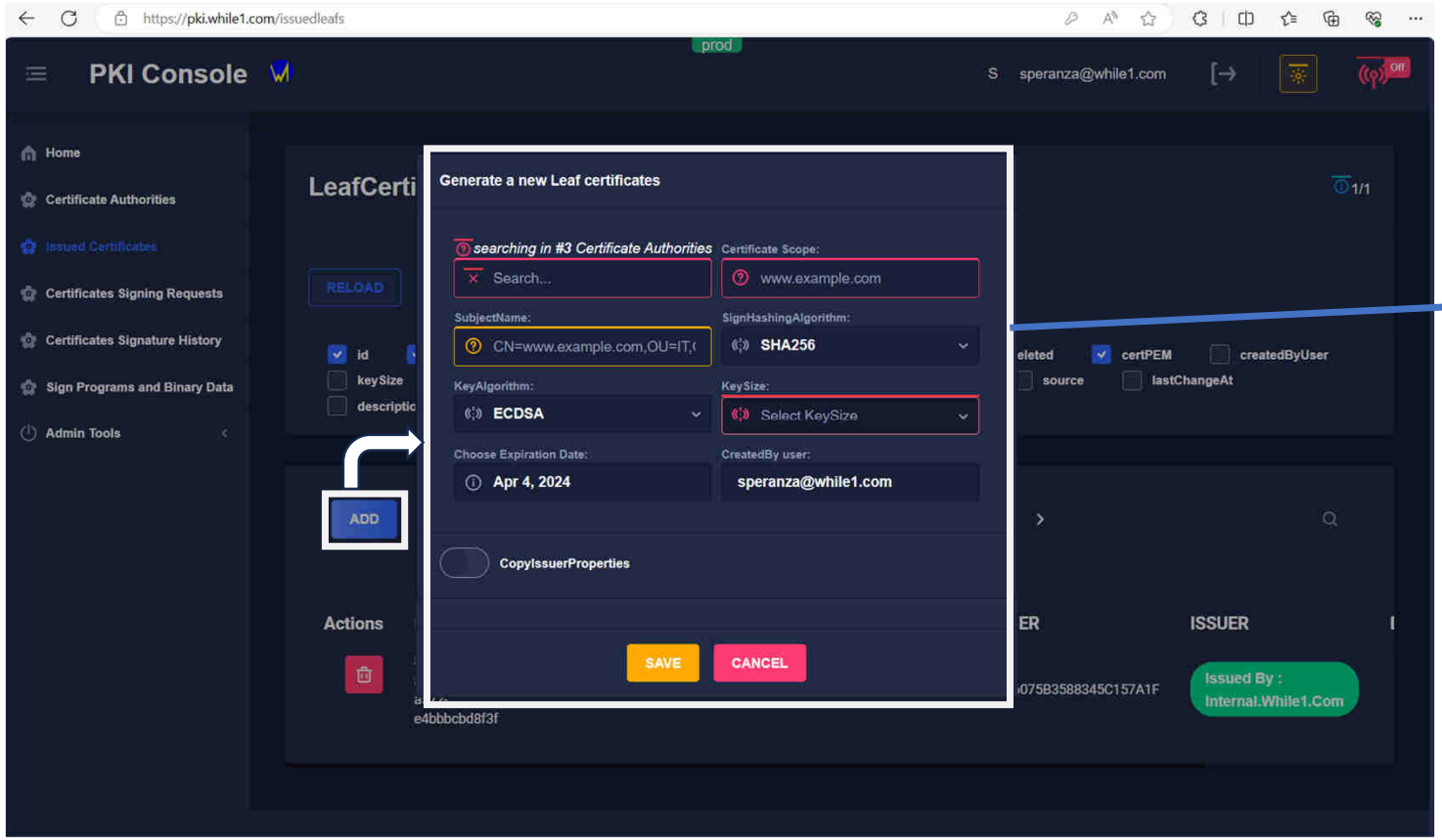
Filter for search. All fields are filterable and produce immediate result

Button to add a leaf certificate. Explain in the next slide

Minimum details for each leaf certificate. The issuer is a intermediate or a root certificate, previously managed.

The delete button will delete the certificate. Since the certificate is a leaf, the deletion will impact and add to CRL this certificate only.

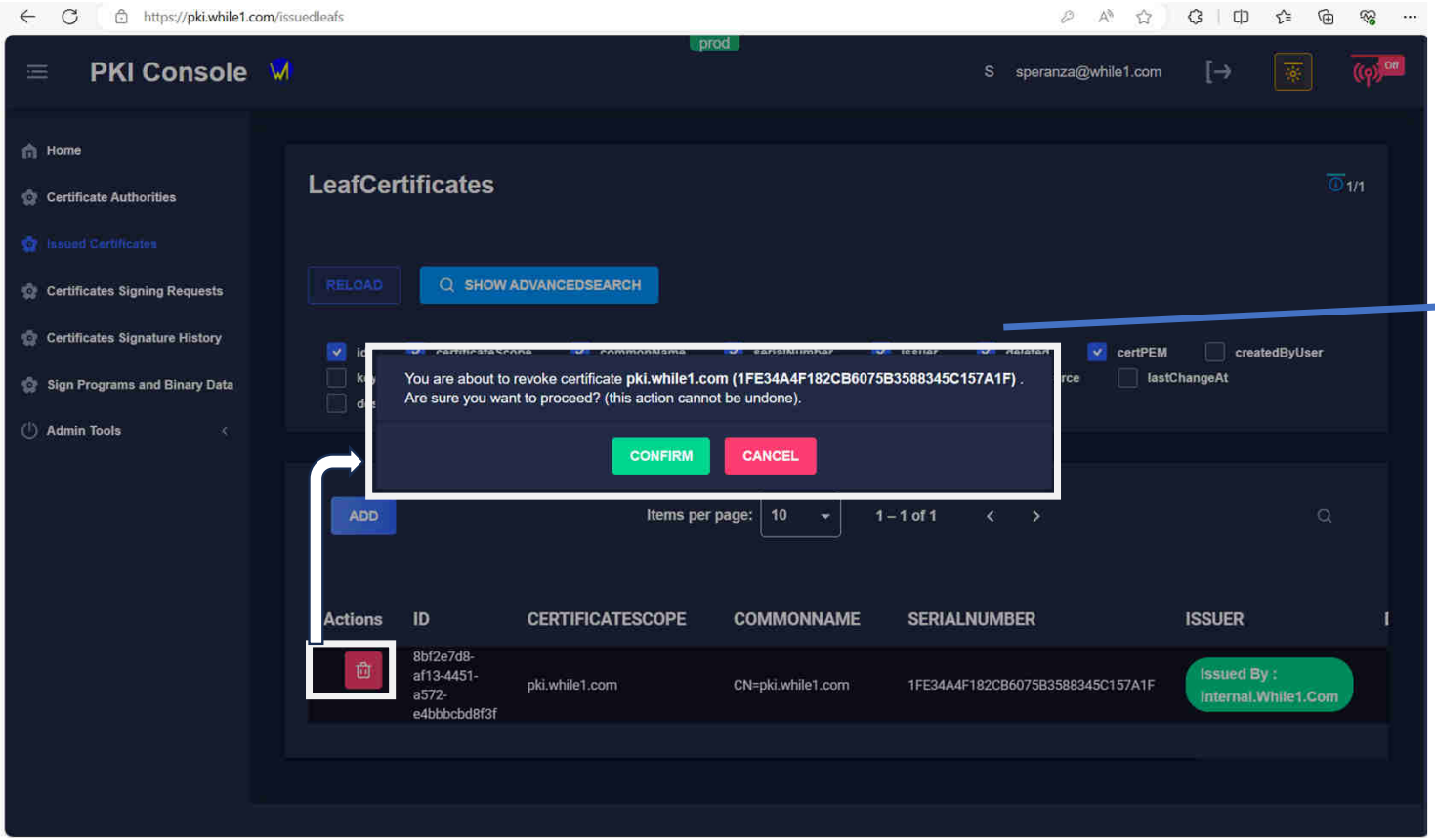
while1.com // PKI >> add new leaf certificate



In this dialog we new leaf certificate.
Properties are:

- The certificate authorities → the root or intermediate certificate selected as signer
- Certificate scope → a label (myleaf.while1.com)
- Subject name → if empty, automatically built, else it can be fully specified
- SignHashingAlgorithm, KeyAlgorith, Keysize → possible combination of valid values and size
- Choose Expiration Date → you can choose the expiration date that best fit the needs. It cannot exceed the signer expiration date.
- Copy Issuer Properties → selecting this feature, all cryptography attributes will be inherited from the signer.

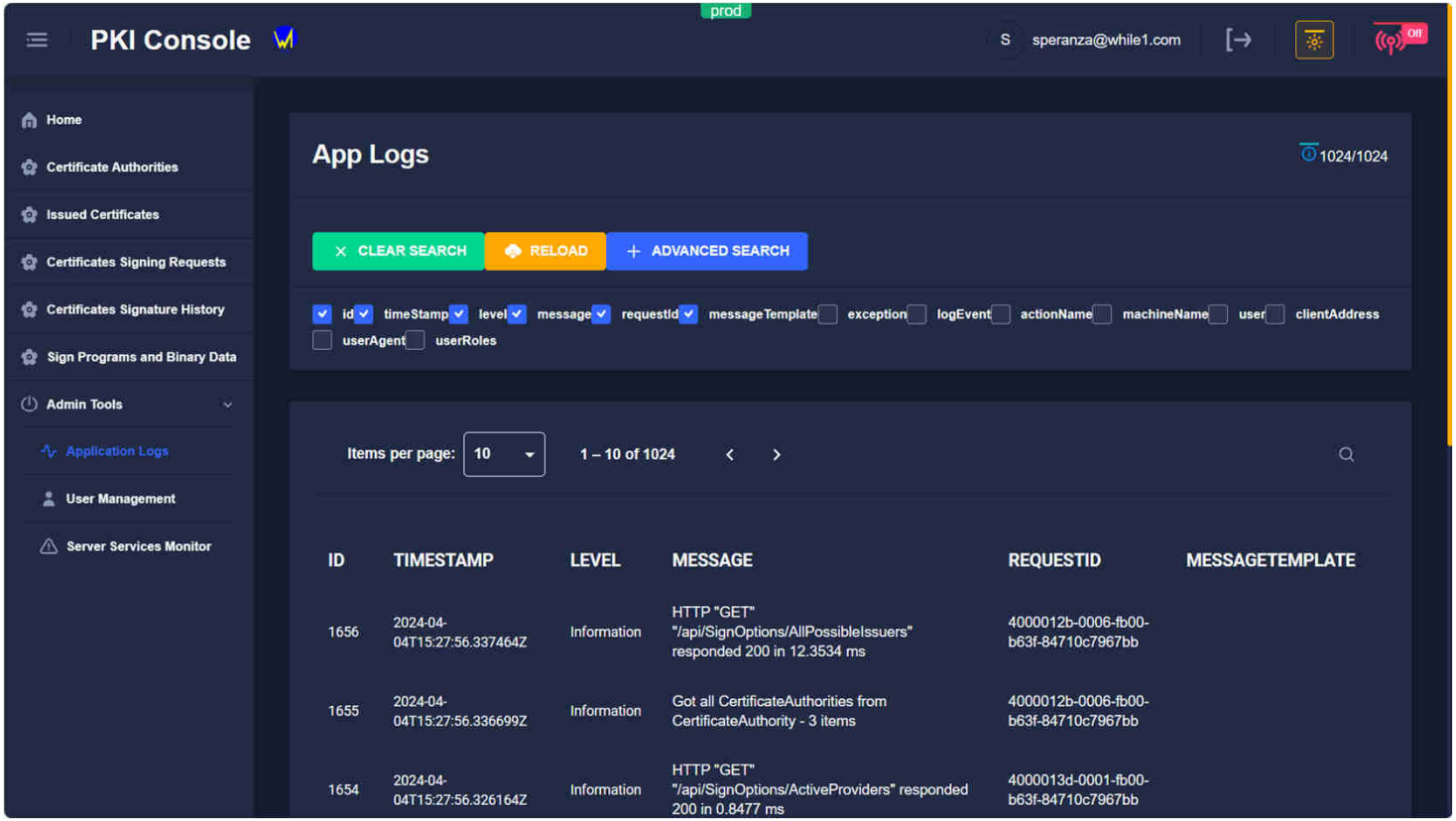
while1.com // PKI >> delete a leaf certificate



In this dialog we can delete a leaf certificate.
The deletion will affect only this certificate. The serial number of the certificate will be automatically inserted in crl file.

A new certificate with same common name can be created again.

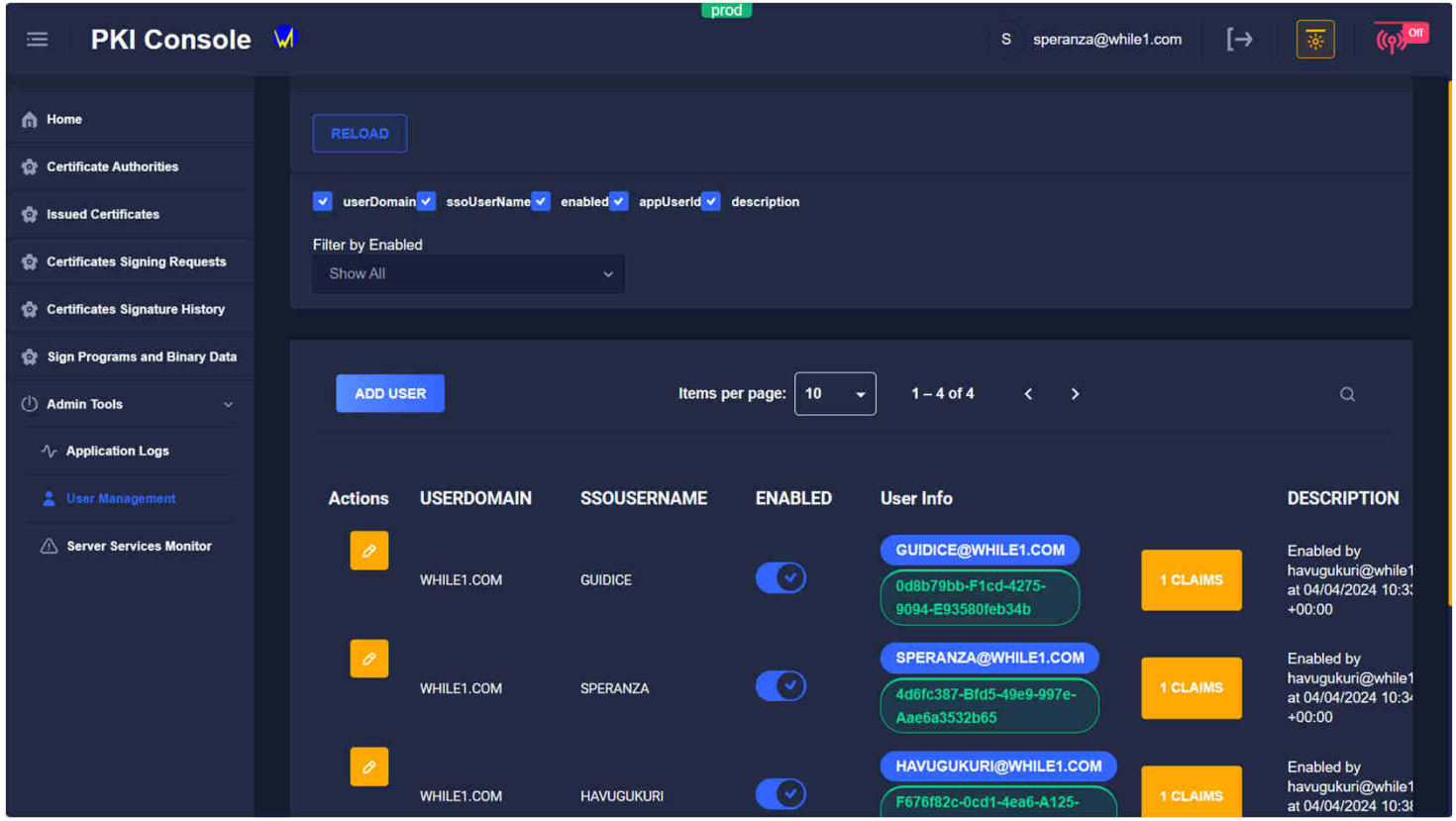
while1.com // PKI >> application logs



In this page, an authorized user can search into logs.

This page in meant for administrative purposes only.

while1.com // PKI >> user management



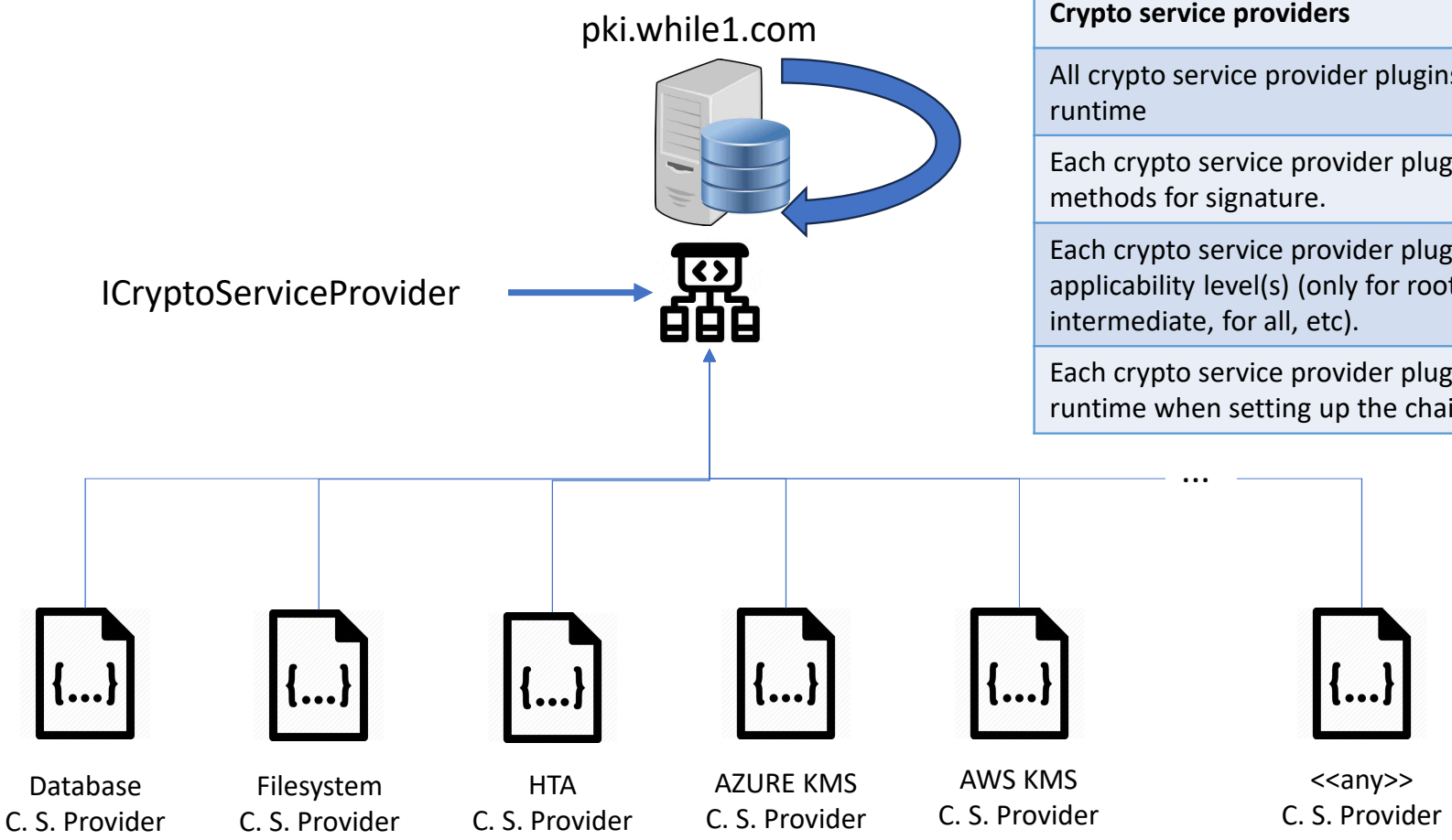
In this page, an authorized user can add or revoke a user.

This page in meant for administrative purposes only.

Hidden api	Scope
submitCSR	This api allows an authorized client to push a CSR and get back a signed certificate. This api allow to specify the provider to be used.
expiringCertificates	This api allows an authorized client to ask for list of certificate that are going to expire in {xx} days.

Project	Internal deployment	Platform
PKI	Yes	Windows / PostgreSQL

Version	Deployed on	Changelog	Confidentiality Level
V0.9.43278	April, the 3°, 2024	First initial release. Local authentication. Root ca, intermediate ca, leafs, choice between diffent alghorithm and different providers	GENERAL BUSINESS



Crypto service providers
All crypto service provider plugins will be loaded runtime
Each crypto service provider plugin will define methods for signature.
Each crypto service provider plugin will define its applicability level(s) (only for root, for root and intermediate, for all, etc).
Each crypto service provider plugin will be usable at runtime when setting up the chain.