



# Hierachy and organisational Structures as understood by



Oliver Welter - welter@tum.de





**Certificate Issuance Workflow in OpenCA** 

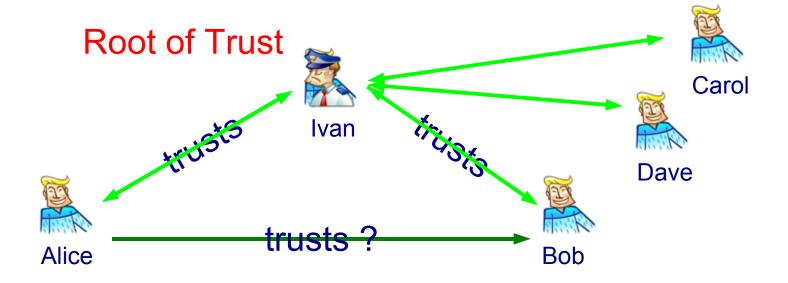
actual Implementation



#### Root of Trust



- Trust: assume that an entity behaves like expected
- transitive Trust-Relation:

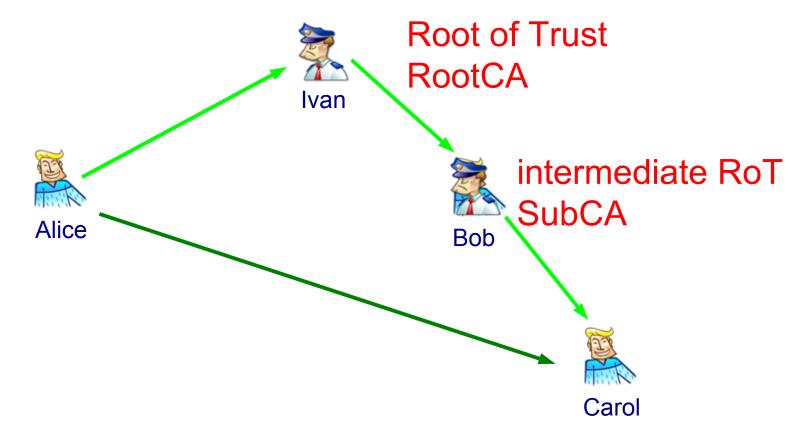




# **Chain of Trust**



#### ★ Chain of Trust

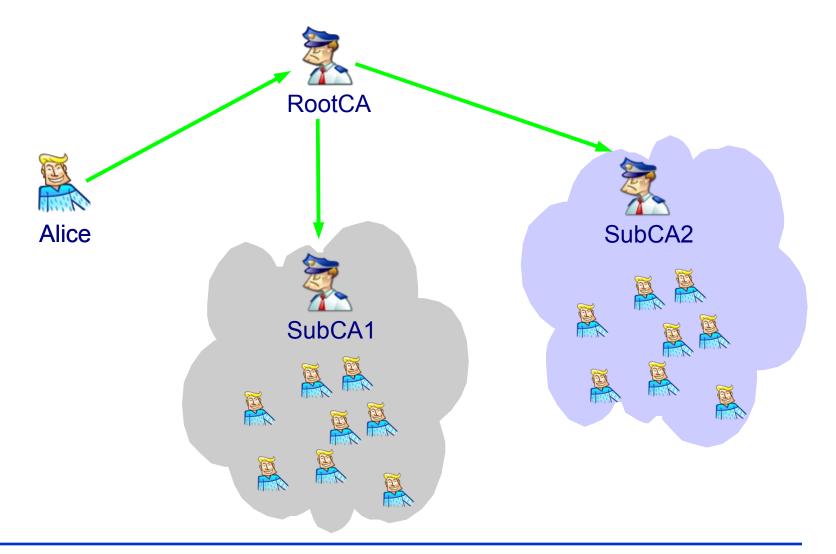




# **Chain of Trust**



# ★ Chain of Trust







**Certificate Issuance Workflow in OpenCA** 

actual Implementation



#### Issue a certificate



- user creates his keypair
- public key e + identity = certificate request (CSR)
- \* "Registration Authority" checks identity
- \* "Registration Authority" approves CSR
  - \* an operator signs the request
- \* "Certification Authority" creates certificate from CSR
- certificate is published



request



identification



certification





# Responsibilities











User

creates

wird benachrichtigt

Registration Officer

identified ?
authorized ?
=> approve
request !

Certification Operator

create certificate





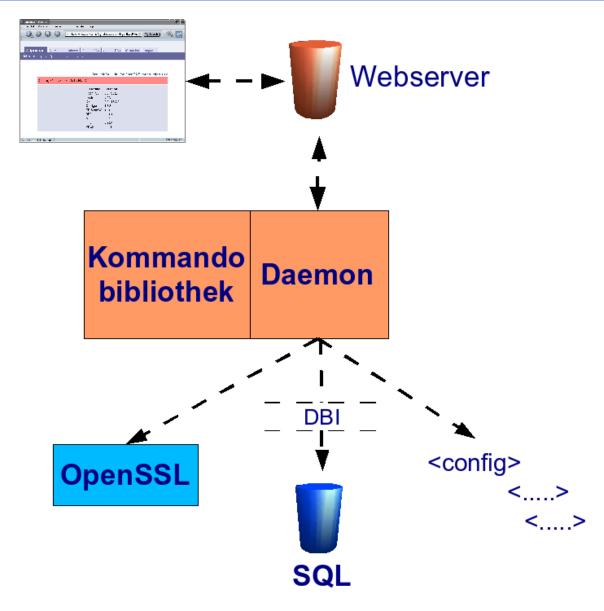
**Certificate Issuance Workflow in OpenCA** 

actual Implementation



#### Architektur

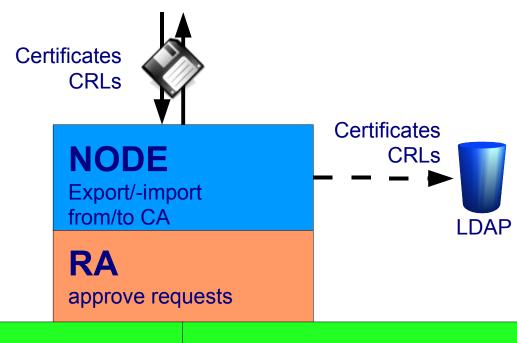






#### **Online-Interfaces**





## **PUB**

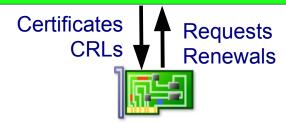
manage own certifocates search others certificates receive information about CA

Certificates CRLs Keys



# **SCEP**

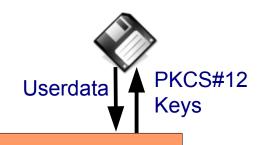
request, renew, download certificates and CRLs via SCEP





#### Offline-Interfaces





## CA

issue/revoke certificates in bulk mode create CRL

## **BATCH**

issue/revoke certificates in batch mode

# **NODE**

Export/-import from/to RA

certificates CRLs







**Certificate Issuance Workflow in OpenCA** 

actual Implementation



#### **Customisation**



- Interface-Logic
  - ★ logical groups of commands
  - ★ separate menu-structure
  - separate access control
- Integration
  - Batch/SCEP allow integration with many third party systems
  - \* starting with upcoming release: powerful shell interface to all components

