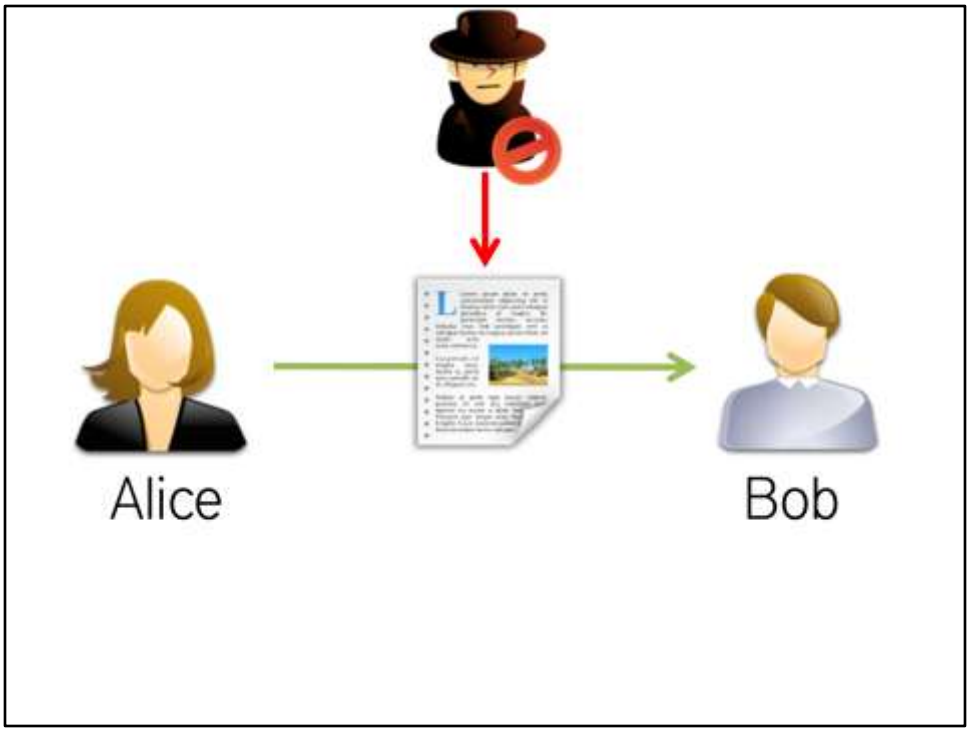
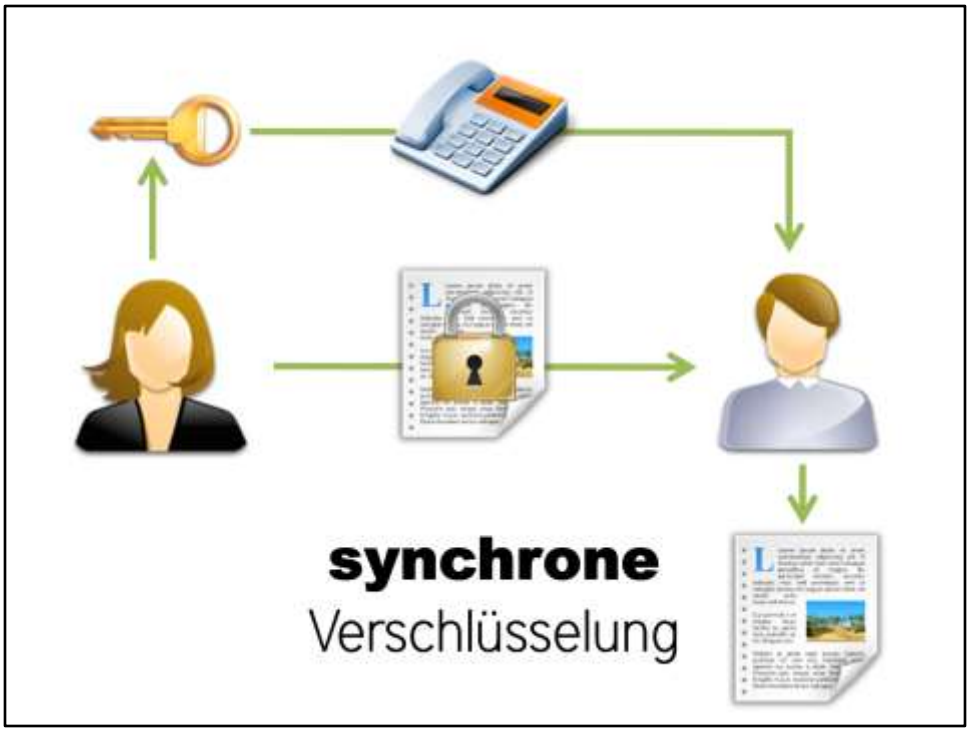
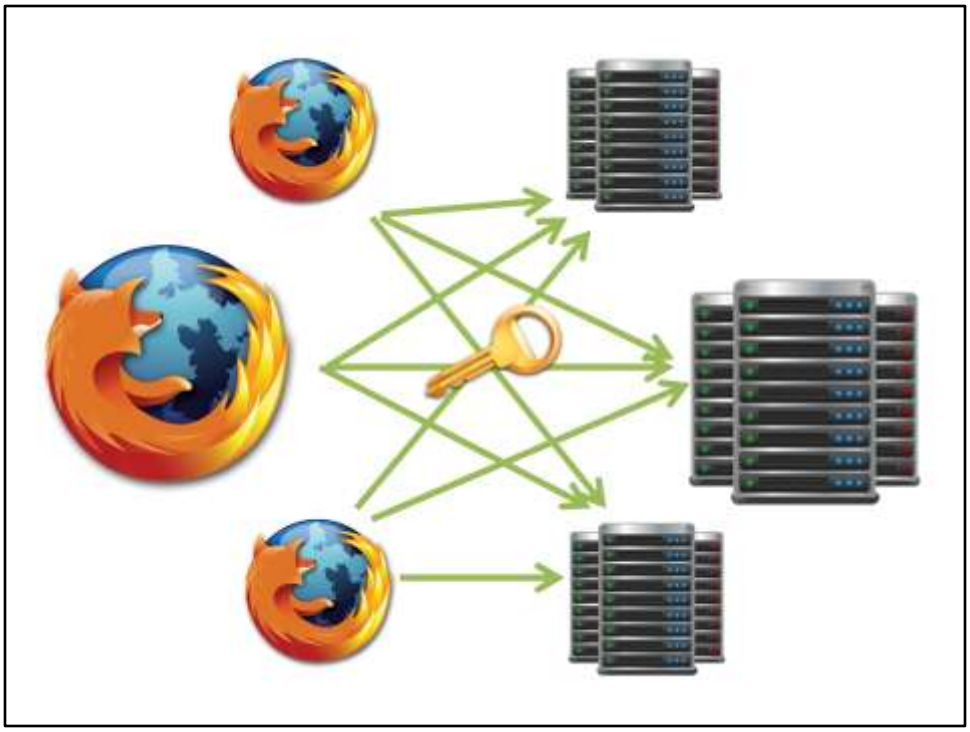
Verschlüsselung



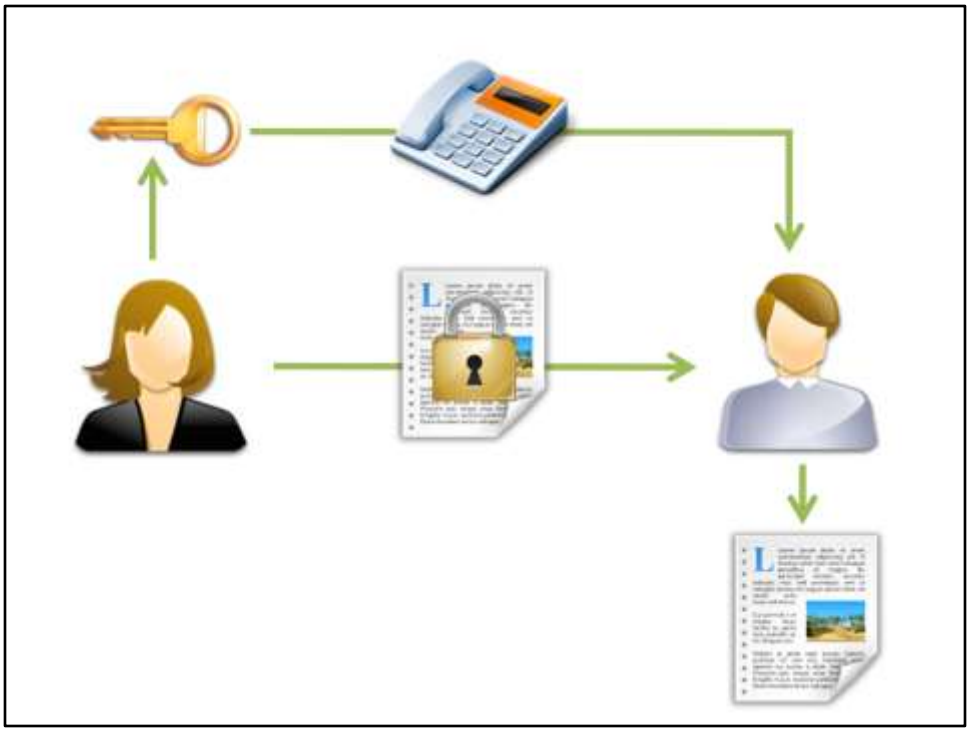
A möchte B eine Nachricht schicken, ohne dass ein Angreifer die Daten lesen kann.



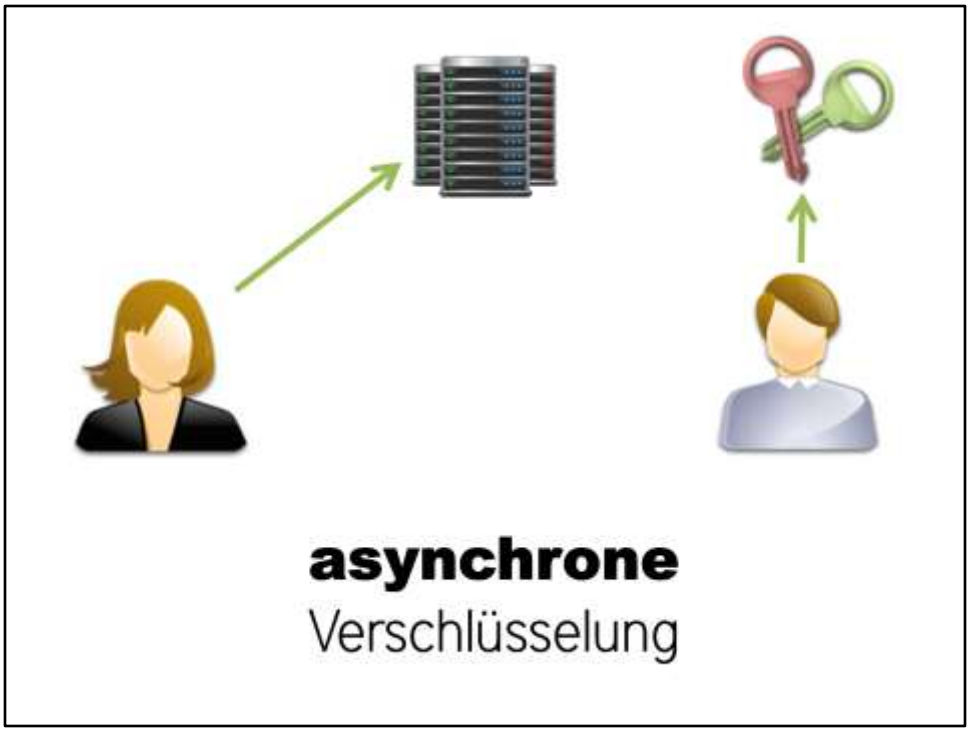
- A erstellt Schlüssel und verschlüsselt Nachricht
- Schlüssel muss auf sicherem Weg an B übertragen werden



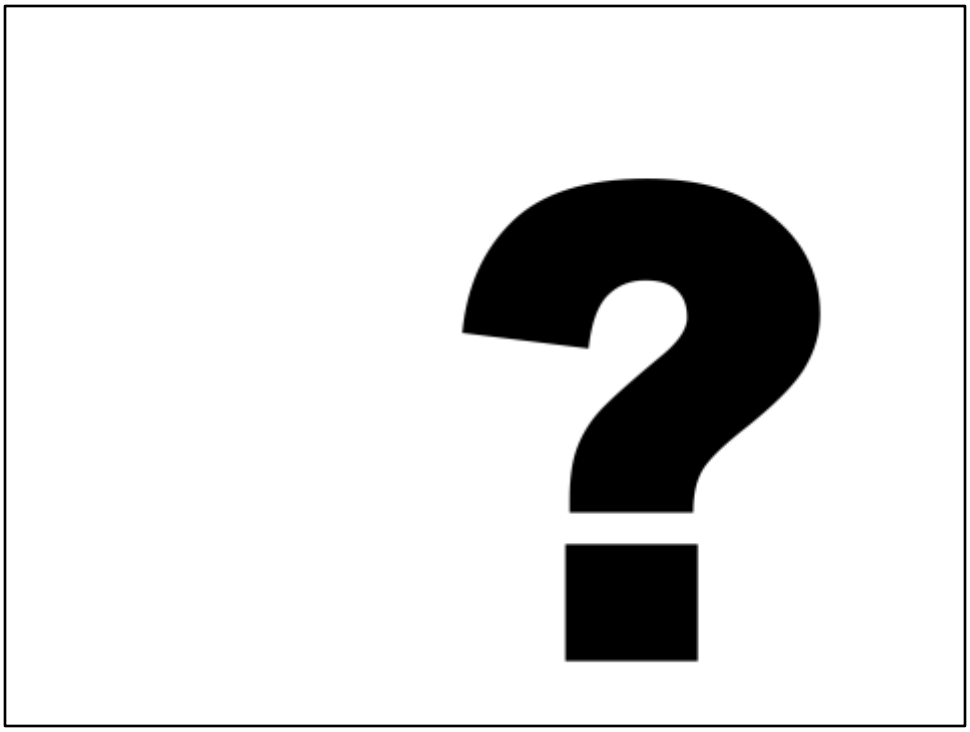
- Problem: Browser kennt viele Adressen, Webserver hat viele Benutzer



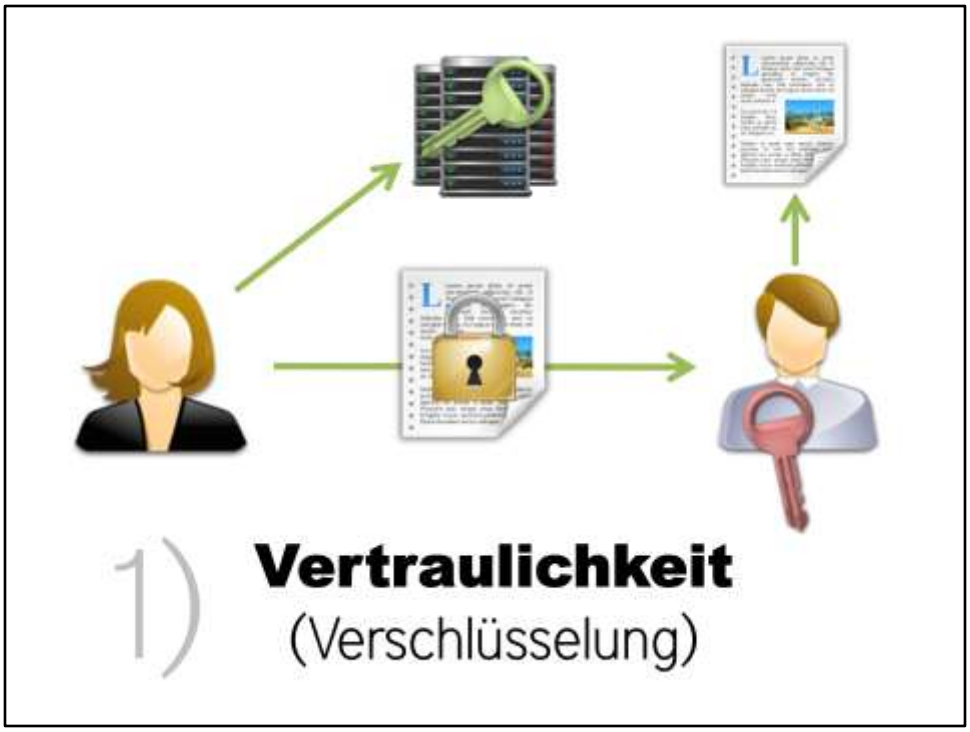
- Verschlüsselung ohne Übertragung eines geheimen Schlüssels nötig



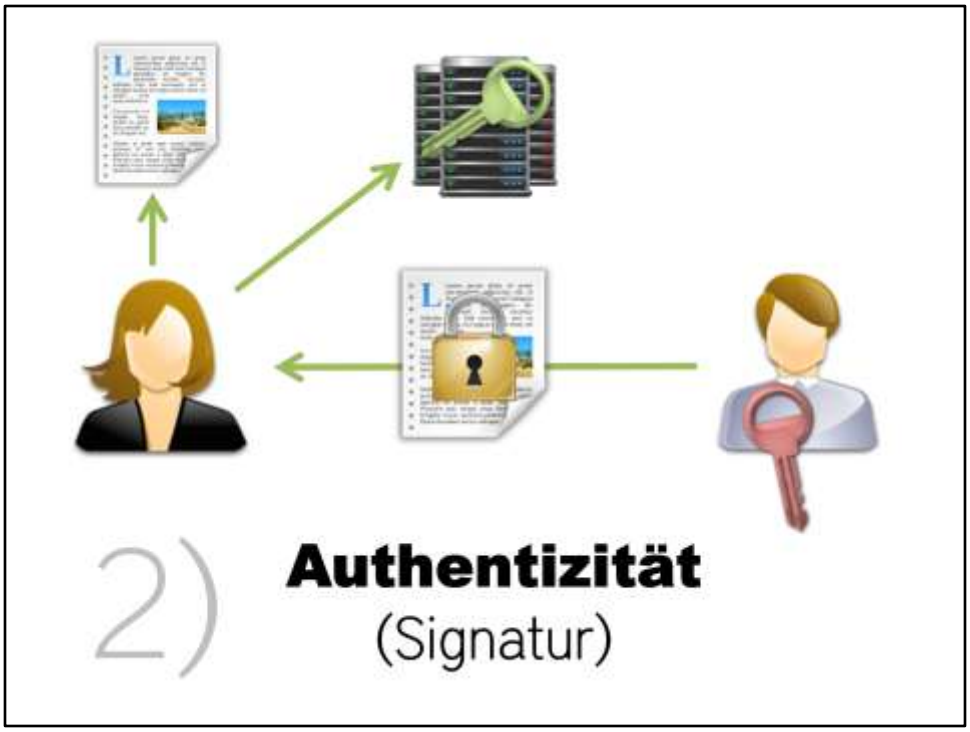
- B generiert Schlüsselpaar
- Privater Schlüssel muss geheim bleiben
- öffentlicher Schlüssel wird bekanntgegeben



Frage an die Zuschauer: Welches Verfahren ist sicherer?

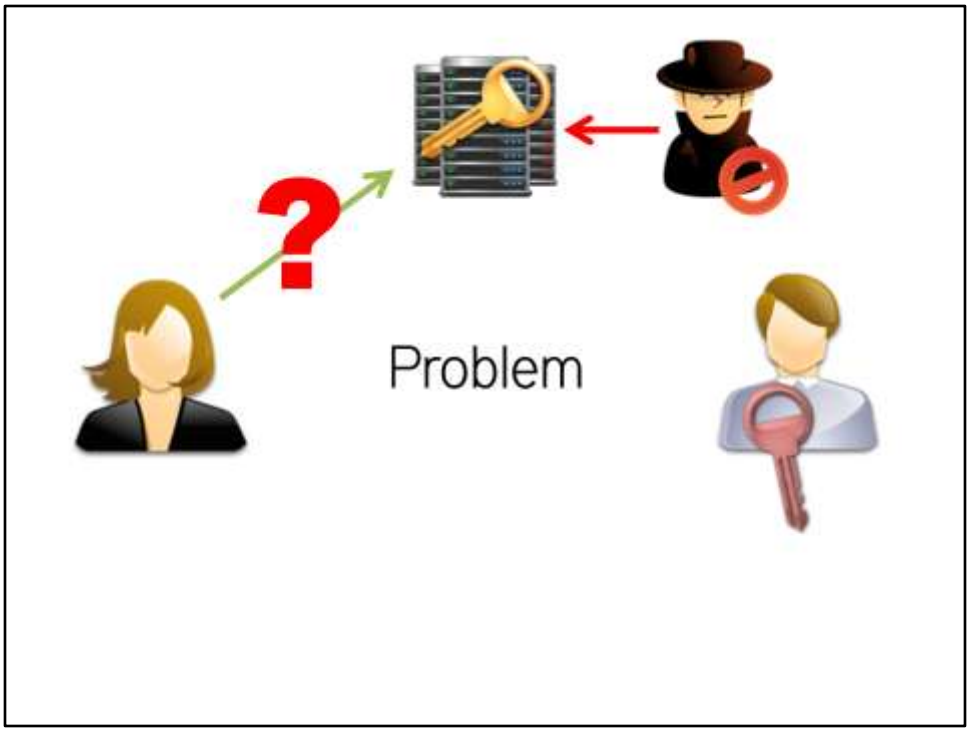


- A verschlüsselt mit PK
- B entschlüsselt mit SK

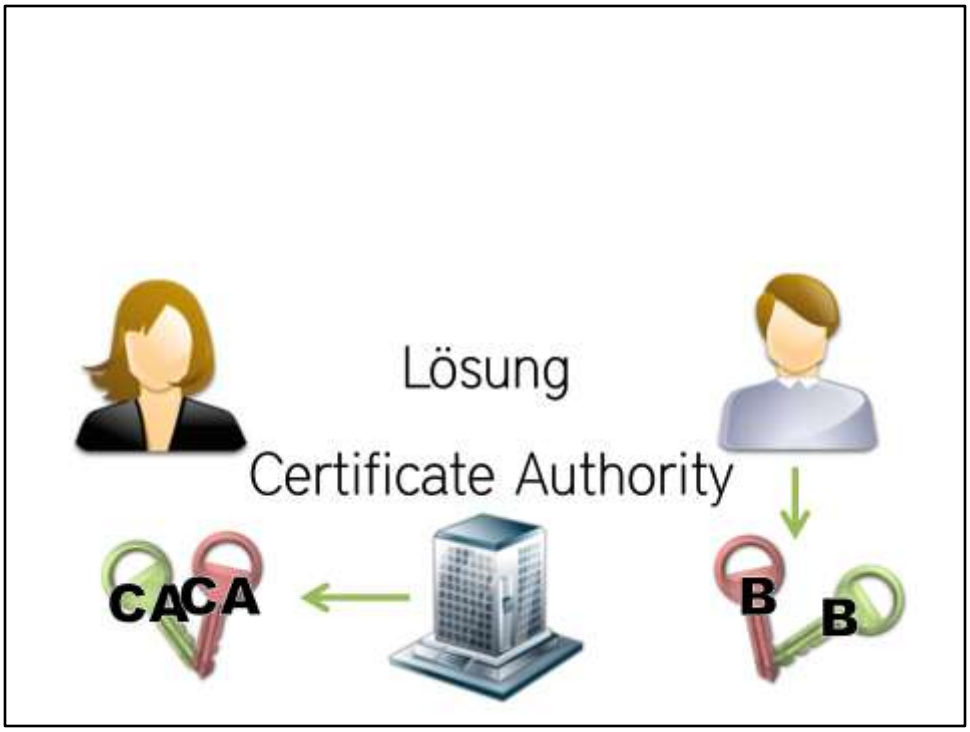


- B verschlüsselt mit SK
- A entschlüsselt mit PK


HTTPS

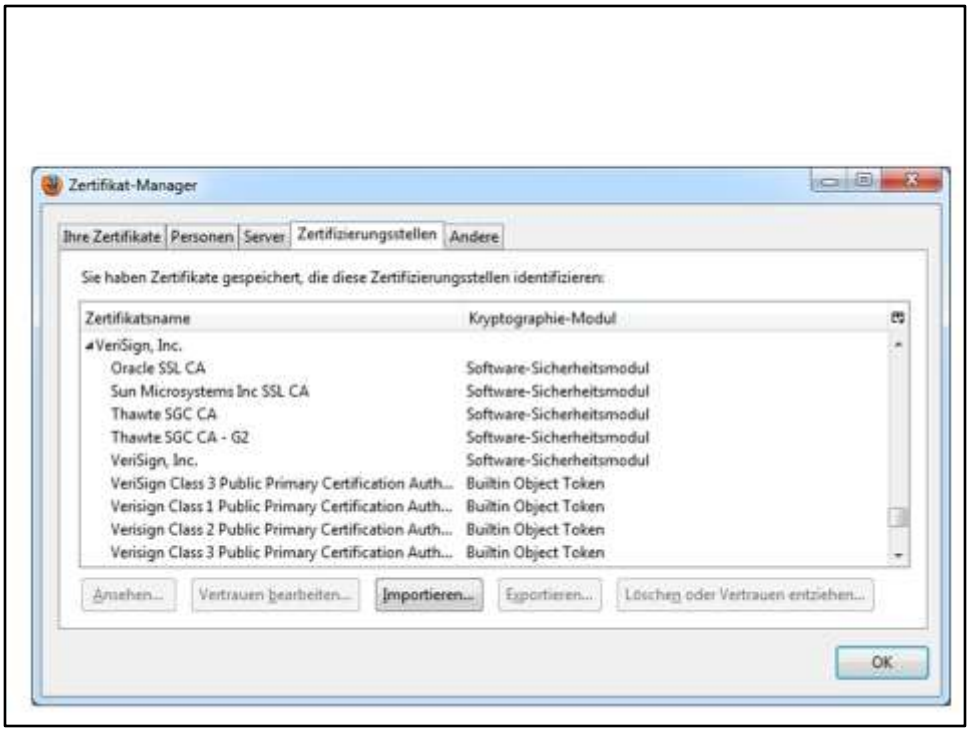


- Wie kann A sicher sein, dass der PK wirklich B gehört und nicht einem Angreifer?

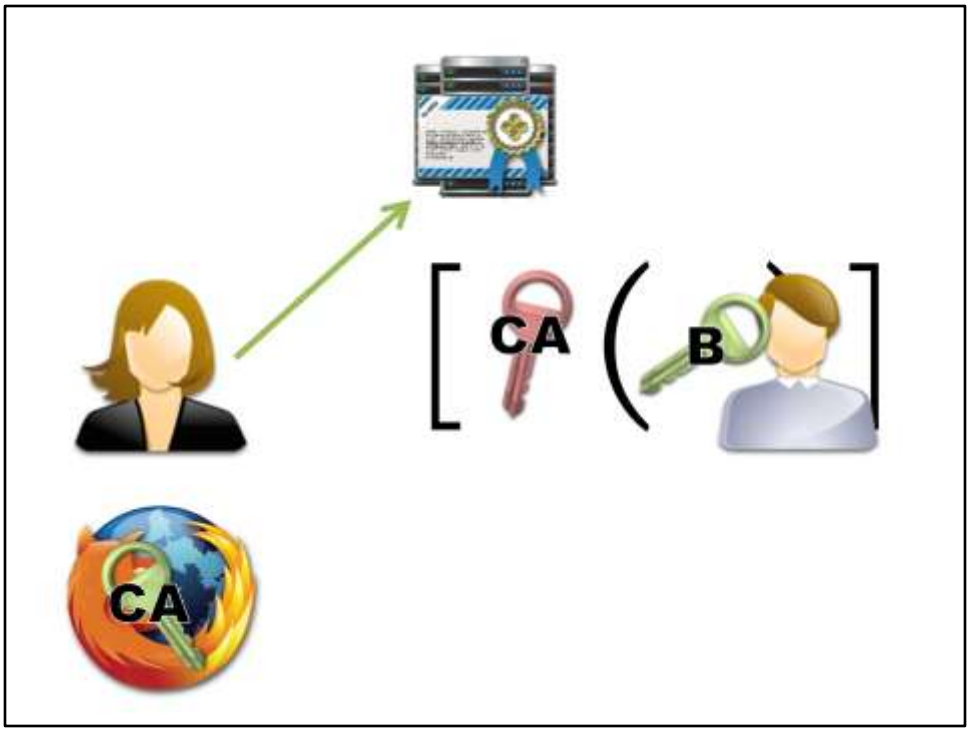


- B und CA generieren Schlüsselpaare
- Private Schlüssel bleiben geheim

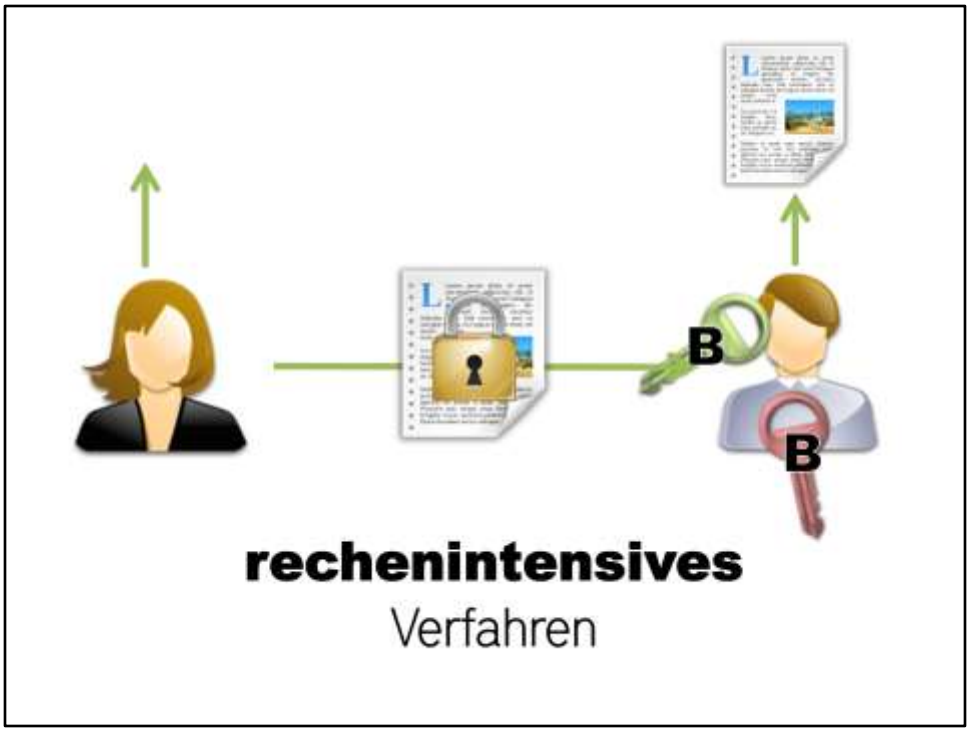
- CA signiert PK von B → Zertifikat
- PK der CA wird im Browser integriert



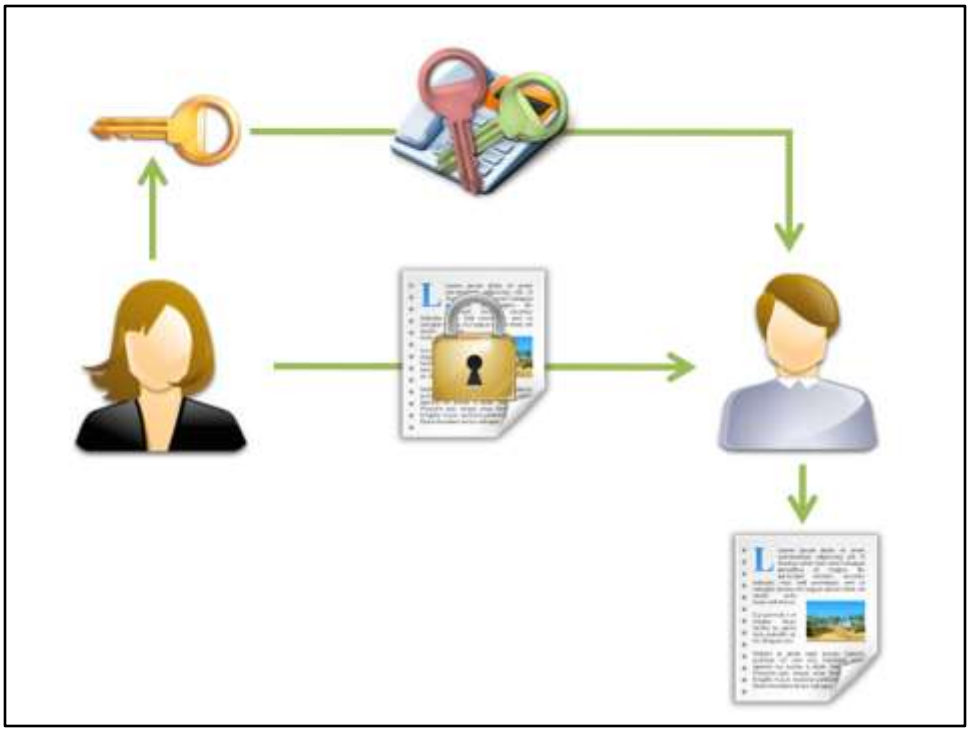
Hier sieht man die vorinstallierten CA-Schlüssel im Firefox



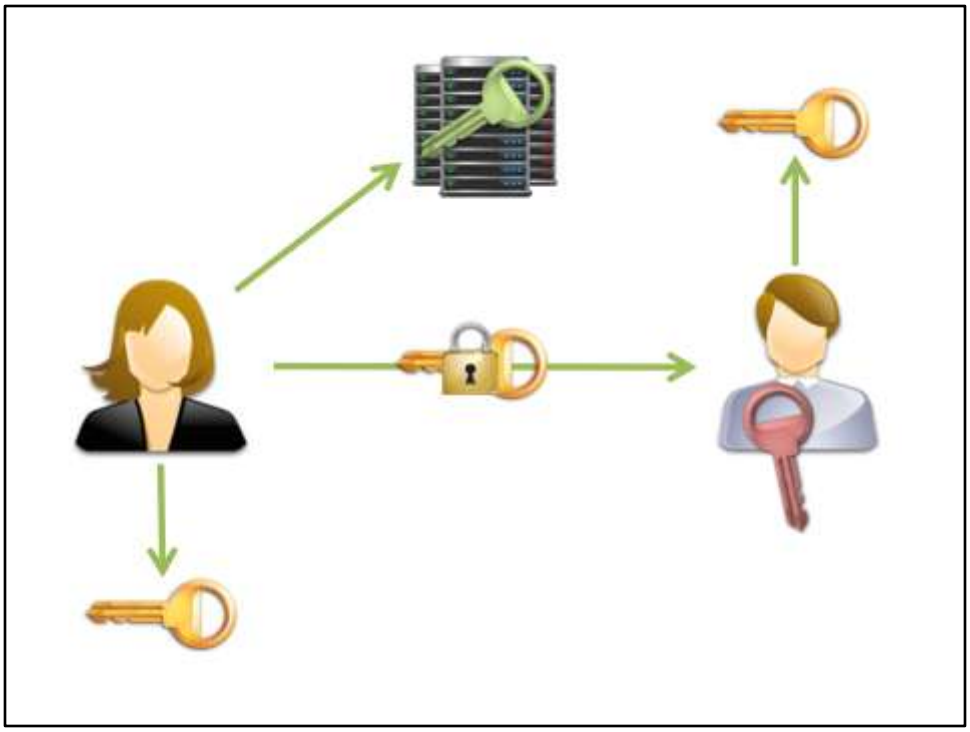
- A hat PK der CA und Zertifikat
- A entschlüsselt PK von B mit PK von CA



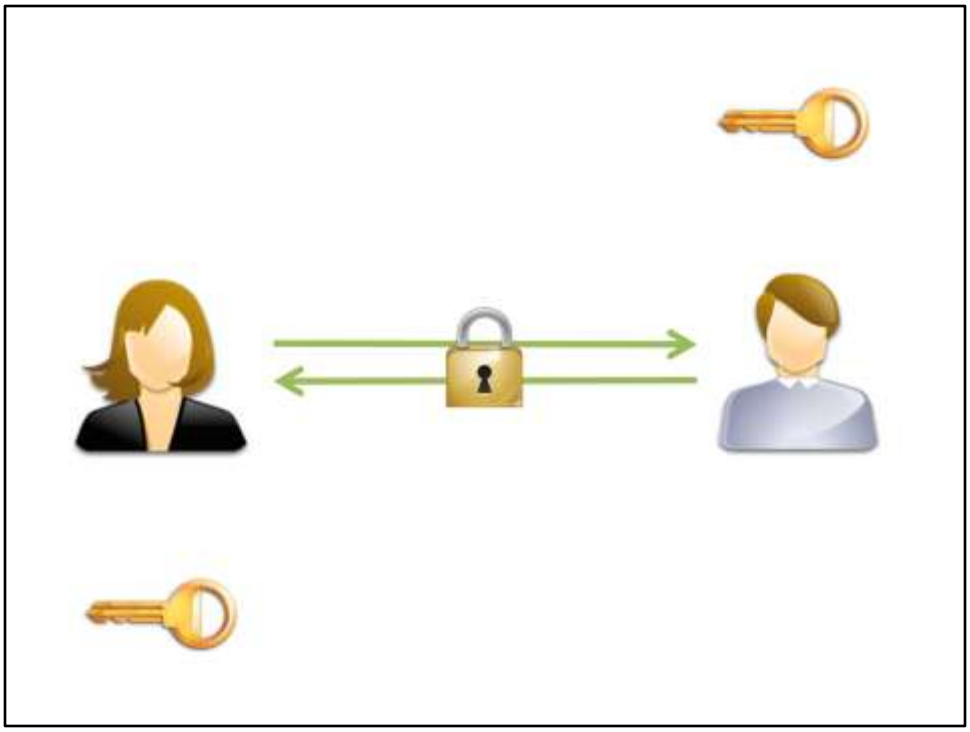
- Nun kann A ganz normal mit dem PK von B an diesen verschlüsselte Nachrichten schicken
- Problem: sehr rechenintensiv, da lange Schlüssel nötig sind



- Daher nutzt HTTPS für die Nutzdaten ein synchrones Verfahren
- Nur für den Schlüsselaustausch wird das asynchrone Verfahren genutzt



- Bei HTTPS wird das asynchrone Verfahren nur genutzt, um einen synchronen Sitzungsschlüssel zu übertragen.



- Danach wird dann nur noch synchron verschlüsselt.