WEB SERVICE SECURITY

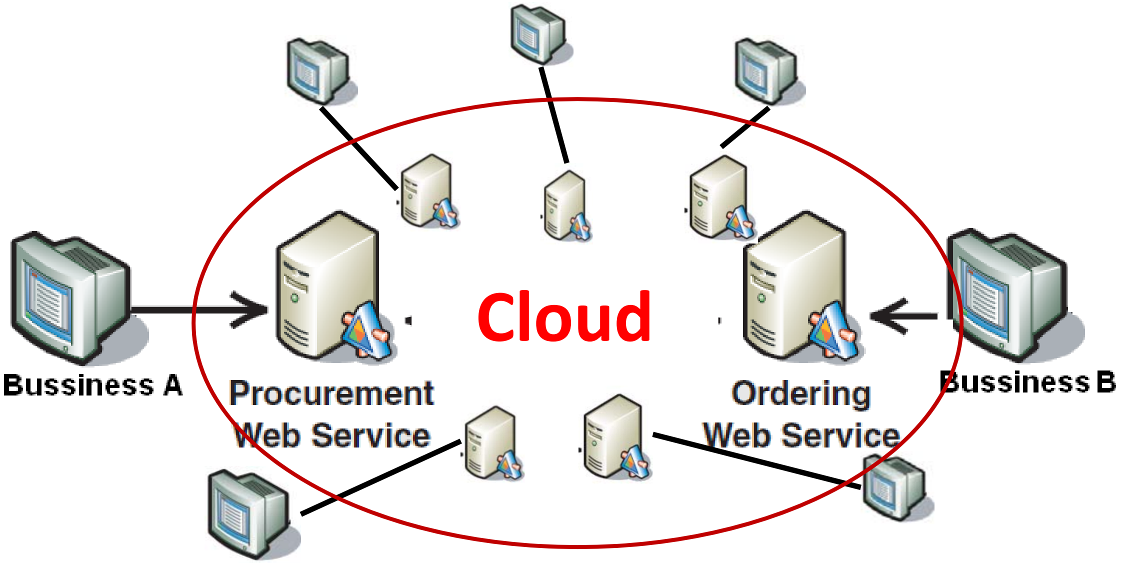
Lan Vu – Shaila Abraham

Outline

1. Introduction
2. Web Service Security Methods
3. Implementation Technology
4. X.509 Model
5. Performance evaluation
6. Conclusion

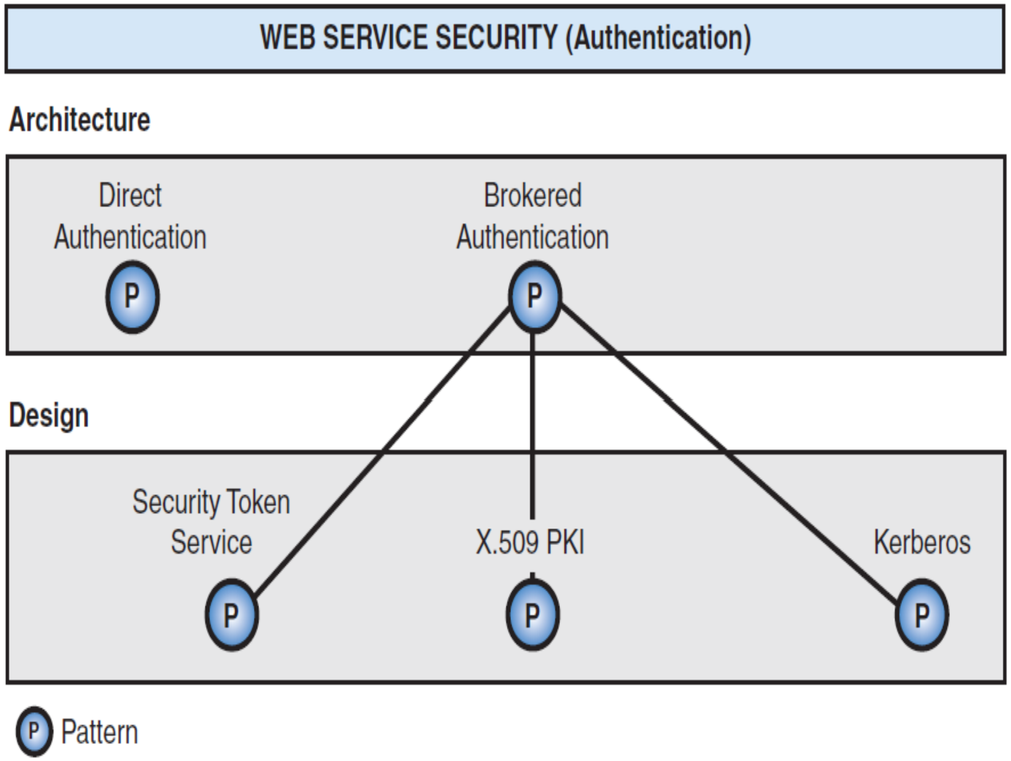
Introduction

- Web Service is a bridge for data sharing and computing between businesses
- A technology enable cloud computing and gain benefits from cloud
- Web services are working in high risk environment
- Without Trust and Security… Web Services are Dead on Arrival



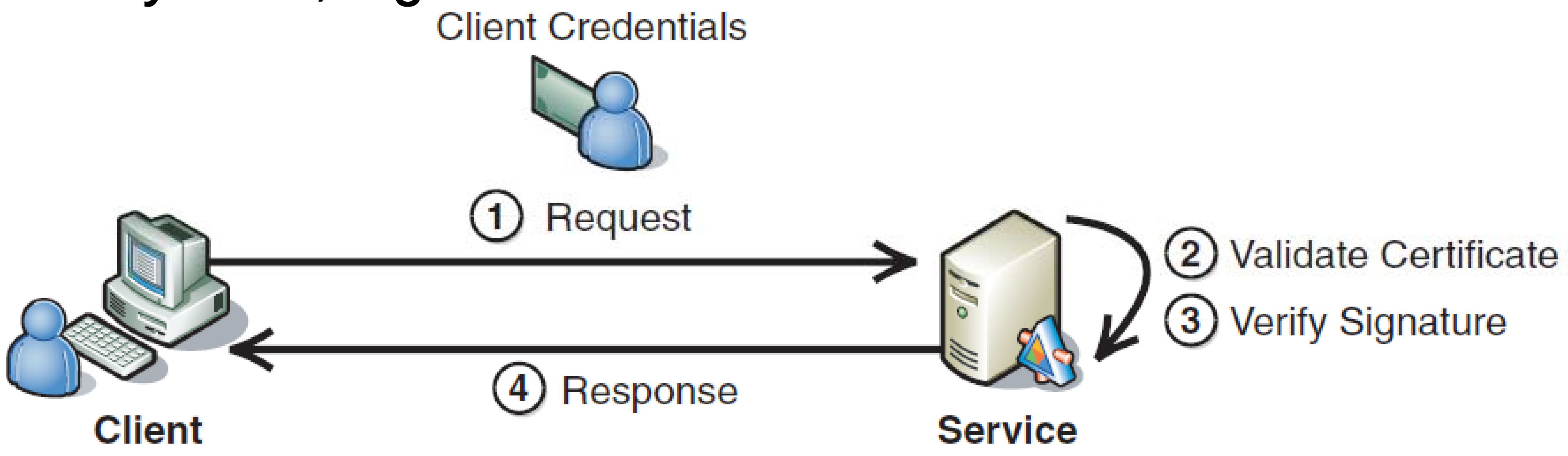
Web Service Security Methods

- Security is related to Authentication , Authorization, Auditing, Confidentiality…
- Our project is about Authentication in Web Service Security
- Authentication Methods:
  - Direct Authentication
  - Brokered Authentication
    - X.509 PKI
    - Kerberos
    - Security Token Service



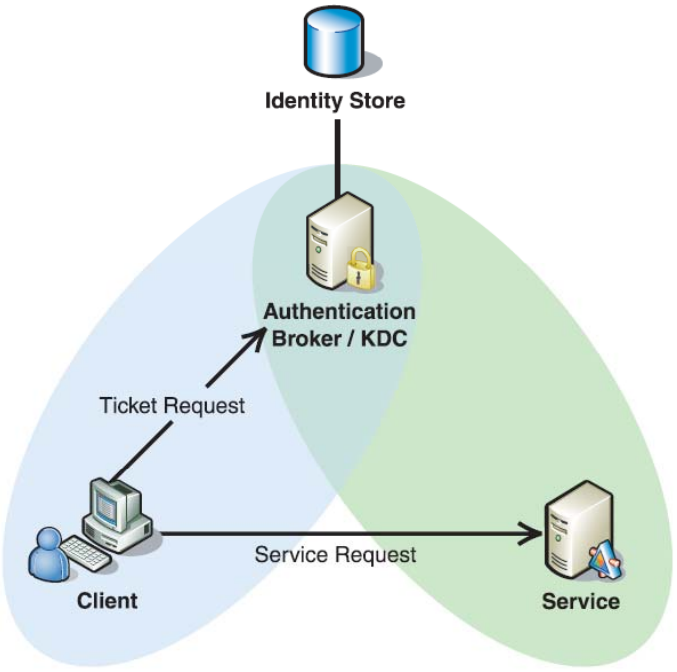
Authentication with X.509 PKI

- Asymmetric Authentication with public key & private key
- Credentials = X.509 certificates
- Certificates are provided by Certificate Authority (CA)
  - Trusted CA: VeriSign, Thawte, and RSA
- Widely used, high cost



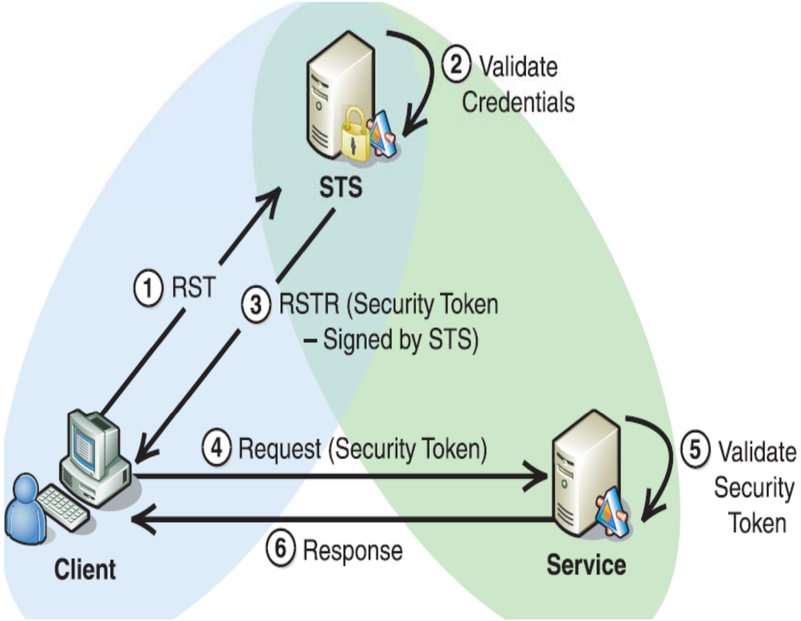
Authentication with Kerberos

- Symmetric Authentication using Kerberos protocol
- Authentication broker is called Key Distribution Center (KDC)
- Lower cost, lower security but better on performance
- Preferred to use for client within a domain



Authentication with Security Token Service

- Authentication broker is an Web service that validate credentials
- Benefits: flexible, enable different authentication protocols
- Performance is better then X.509 but lower than Kerberos


STS
② Validate Credentials
① RST
③ RSTR (Security Token – Signed by STS)
④ Request (Security Token)
⑤ Validate Security Token
⑥ Response
Client
Service

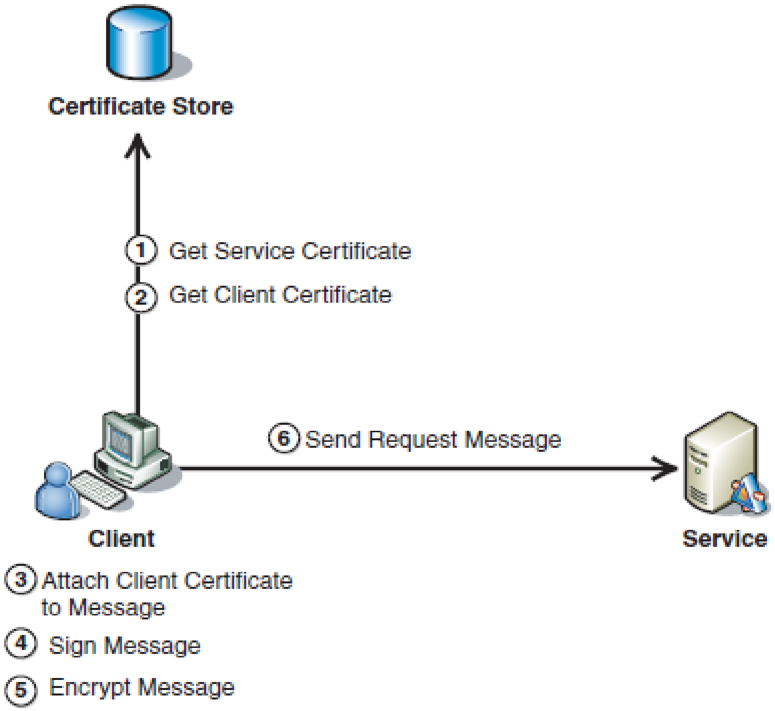
Implementation Technology

- WSE 3.0 (Web Service Enhancements 3.0): Security Product for Web Services.
  - Used to build secure web services easily.
  - Turnkey Security Scenarios (Username over Certificate, MutualCertifcate, Kerberos Certificate).
  - ASMX + WSE 3.0 builds a secure web service.

- WCF (Windows Communication Foundation): Programming model over SOA
  - Interoperable with Pre-WCF Technologies
  - Implements web service technologies specified by WS-* specification (WS-Security, WS-Trust, WS-SecureConversation, WS-ReliableMessaging, WS-Coordination, WS-AtomicTransaction)

X.509 Model

- Client initializes and sends a message with X.509 certificate info.
- Service authenticates the client using the X.509 certificate and signature.



Certificate Store

1 Get Service Certificate

2 Get Client Certificate

6 Send Request Message

Client

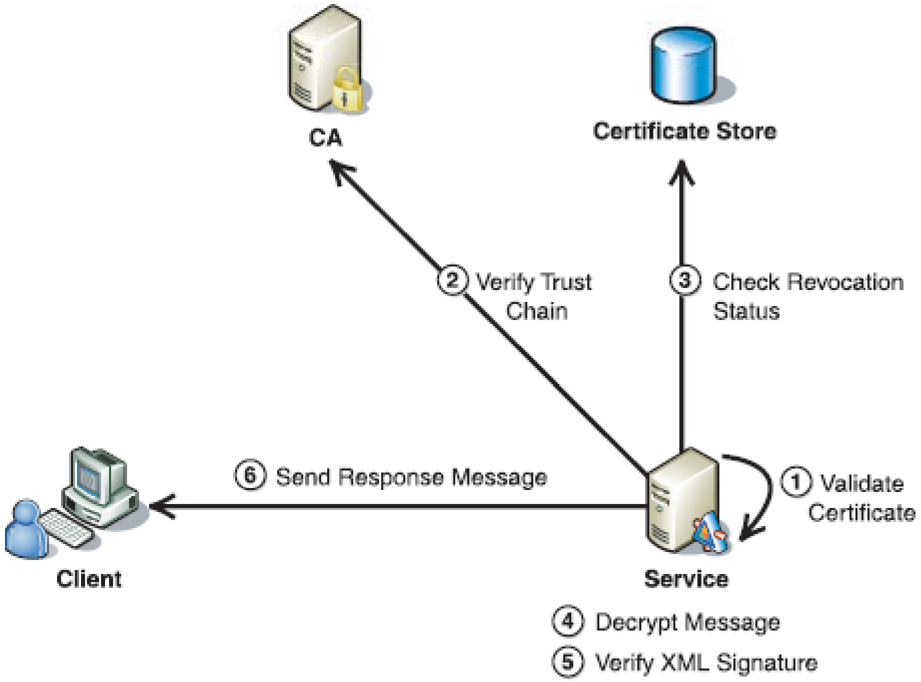
Service

3 Attach Client Certificate to Message

4 Sign Message

5 Encrypt Message

Client Authentication



CA

Certificate Store

2 Verify Trust Chain

3 Check Revocation Status

6 Send Response Message

1 Validate Certificate

Client

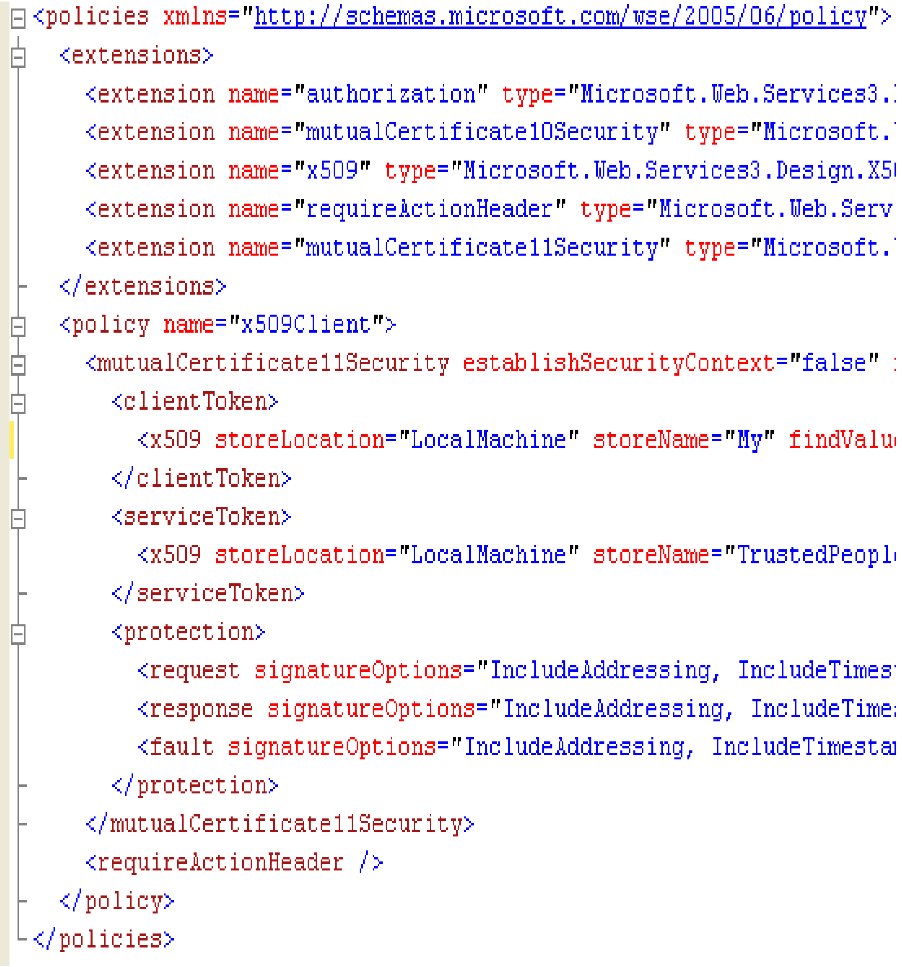
Service

4 Decrypt Message

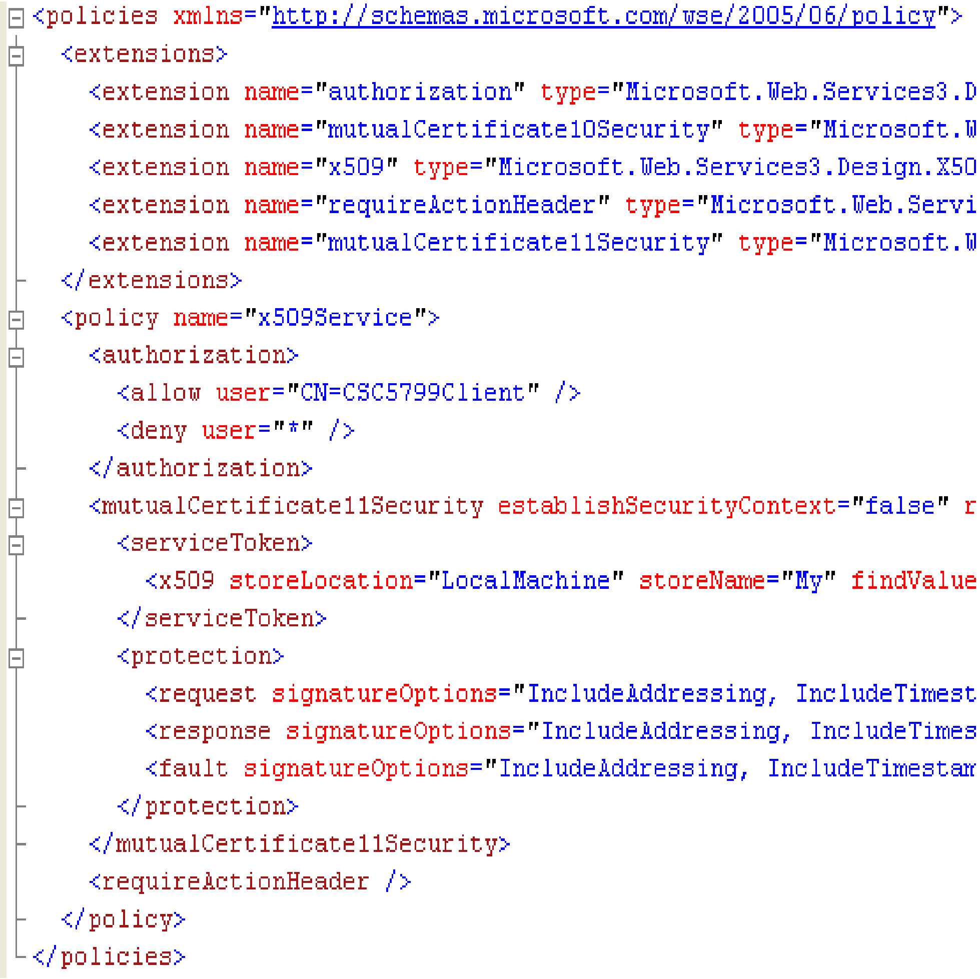
5 Verify XML Signature

Server Authentication

X.509 Sample Code



```
<policies xmlns="http://schemas.microsoft.com/wse/2005/06/policy">
  <extensions>
    <extension name="authorization" type="Microsoft.Web.Services3.
    <extension name="mutualCertificate10Security" type="Microsoft.
    <extension name="x509" type="Microsoft.Web.Services3.Design.X5
    <extension name="requireActionHeader" type="Microsoft.Web.Serv
    <extension name="mutualCertificate11Security" type="Microsoft.
  </extensions>
  <policy name="x509Client">
    <mutualCertificate11Security establishSecurityContext="false"
      <clientToken>
        <x509 storeLocation="LocalMachine" storeName="My" findValu
      </clientToken>
      <serviceToken>
        <x509 storeLocation="LocalMachine" storeName="TrustedPeopl
      </serviceToken>
      <protection>
        <request signatureOptions="IncludeAddressing, IncludeTimes
        <response signatureOptions="IncludeAddressing, IncludeTime
        <fault signatureOptions="IncludeAddressing, IncludeTimesta
      </protection>
    </mutualCertificate11Security>
    <requireActionHeader />
  </policy>
</policies>
```

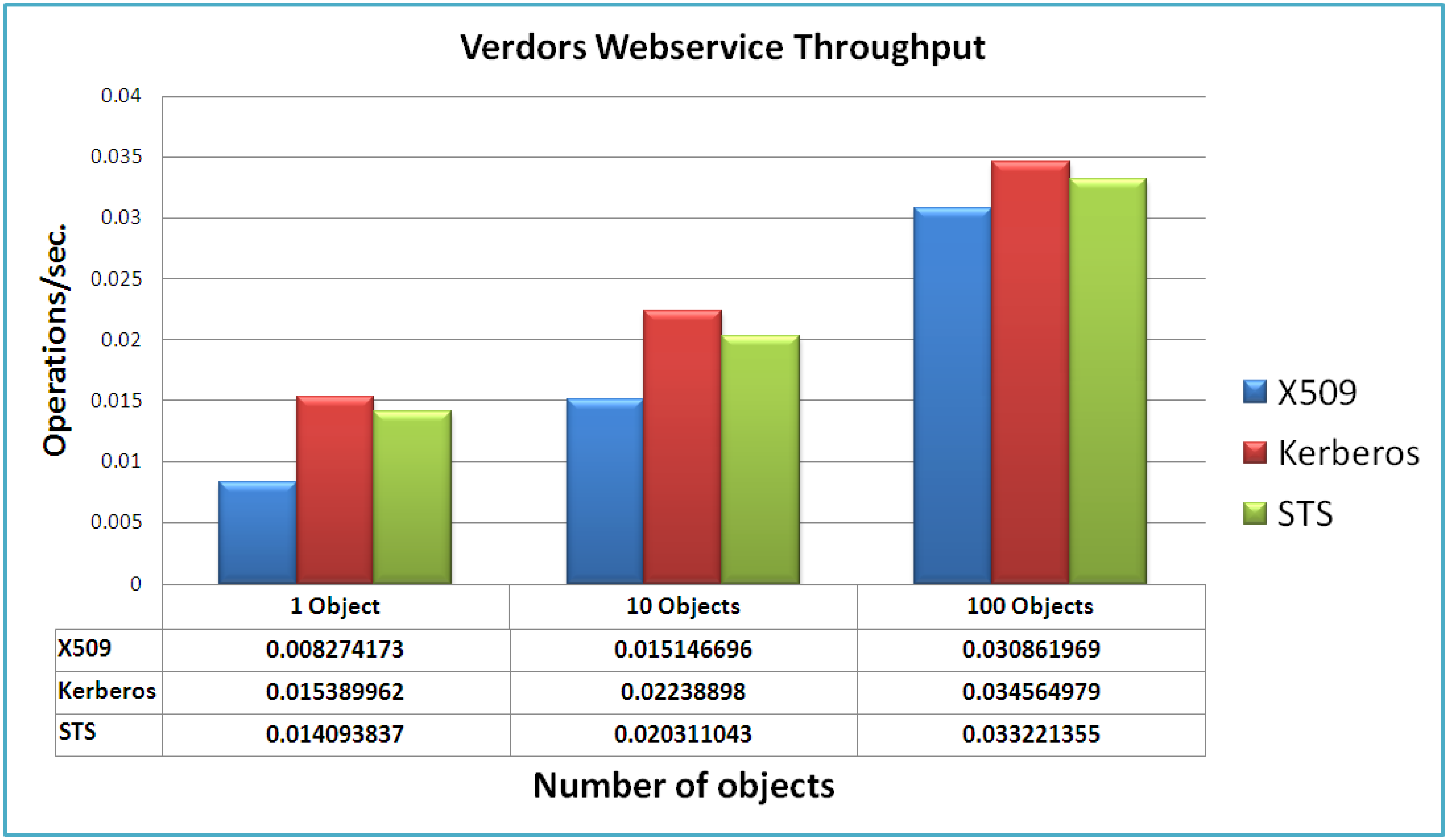


```
<policies xmlns="http://schemas.microsoft.com/wse/2005/06/policy">
  <extensions>
    <extension name="authorization" type="Microsoft.Web.Services3.D
    <extension name="mutualCertificate10Security" type="Microsoft.W
    <extension name="x509" type="Microsoft.Web.Services3.Design.X50
    <extension name="requireActionHeader" type="Microsoft.Web.Servi
    <extension name="mutualCertificate11Security" type="Microsoft.W
  </extensions>
  <policy name="x509Service">
    <authorization>
      <allow user="CN=CSC5799Client" />
      <deny user="*" />
    </authorization>
    <mutualCertificate11Security establishSecurityContext="false" r
      <serviceToken>
        <x509 storeLocation="LocalMachine" storeName="My" findValue
      </serviceToken>
      <protection>
        <request signatureOptions="IncludeAddressing, IncludeTimest
        <response signatureOptions="IncludeAddressing, IncludeTimes
        <fault signatureOptions="IncludeAddressing, IncludeTimestam
      </protection>
    </mutualCertificate11Security>
    <requireActionHeader />
  </policy>
</policies>
```

Client Policy File

Server Policy File

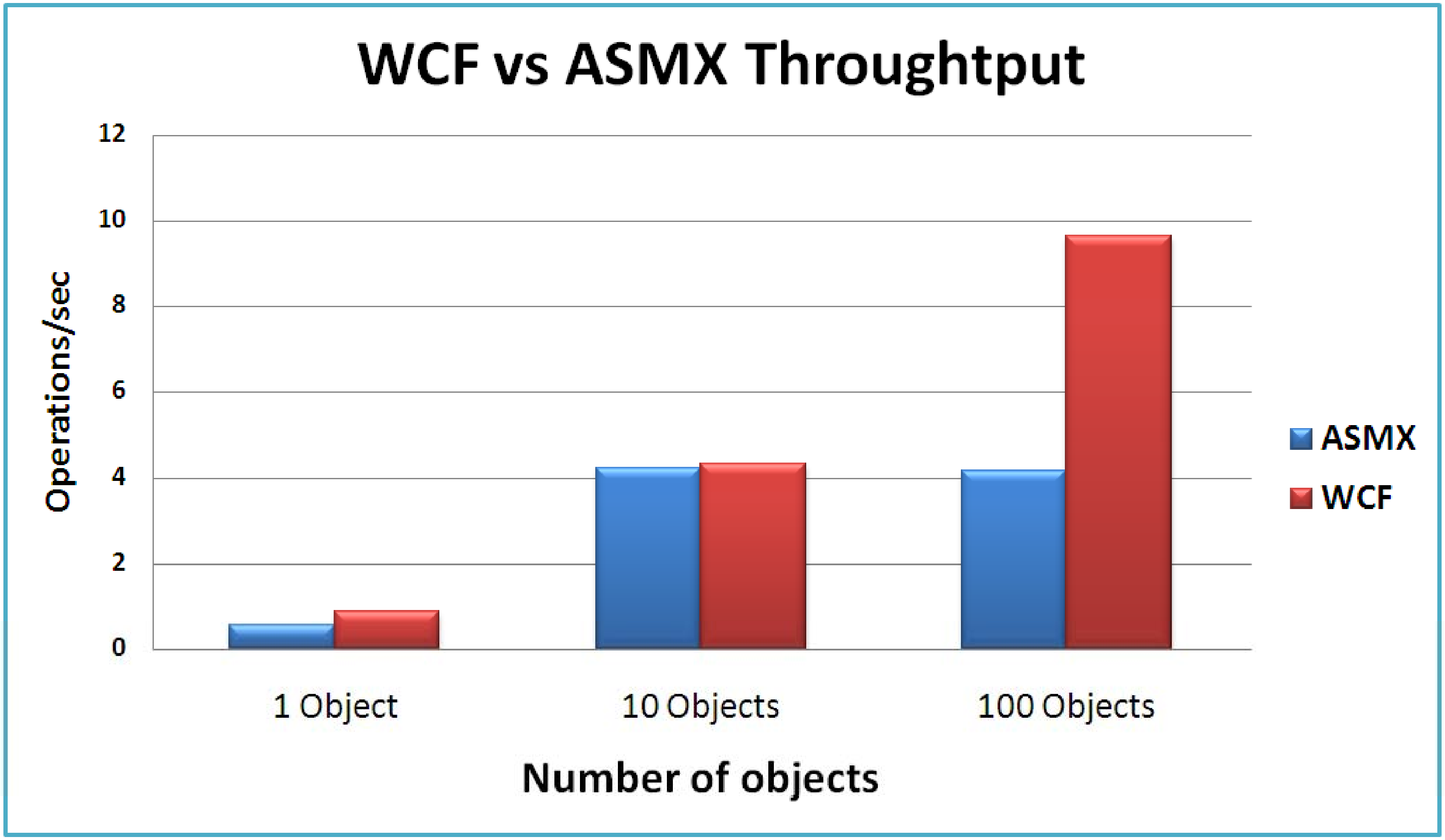
Performance Evaluation



	1 Object	10 Objects	100 Objects
X509	0.008274173	0.015146696	0.030861969
Kerberos	0.015389962	0.02238898	0.034564979
STS	0.014093837	0.020311043	0.033221355

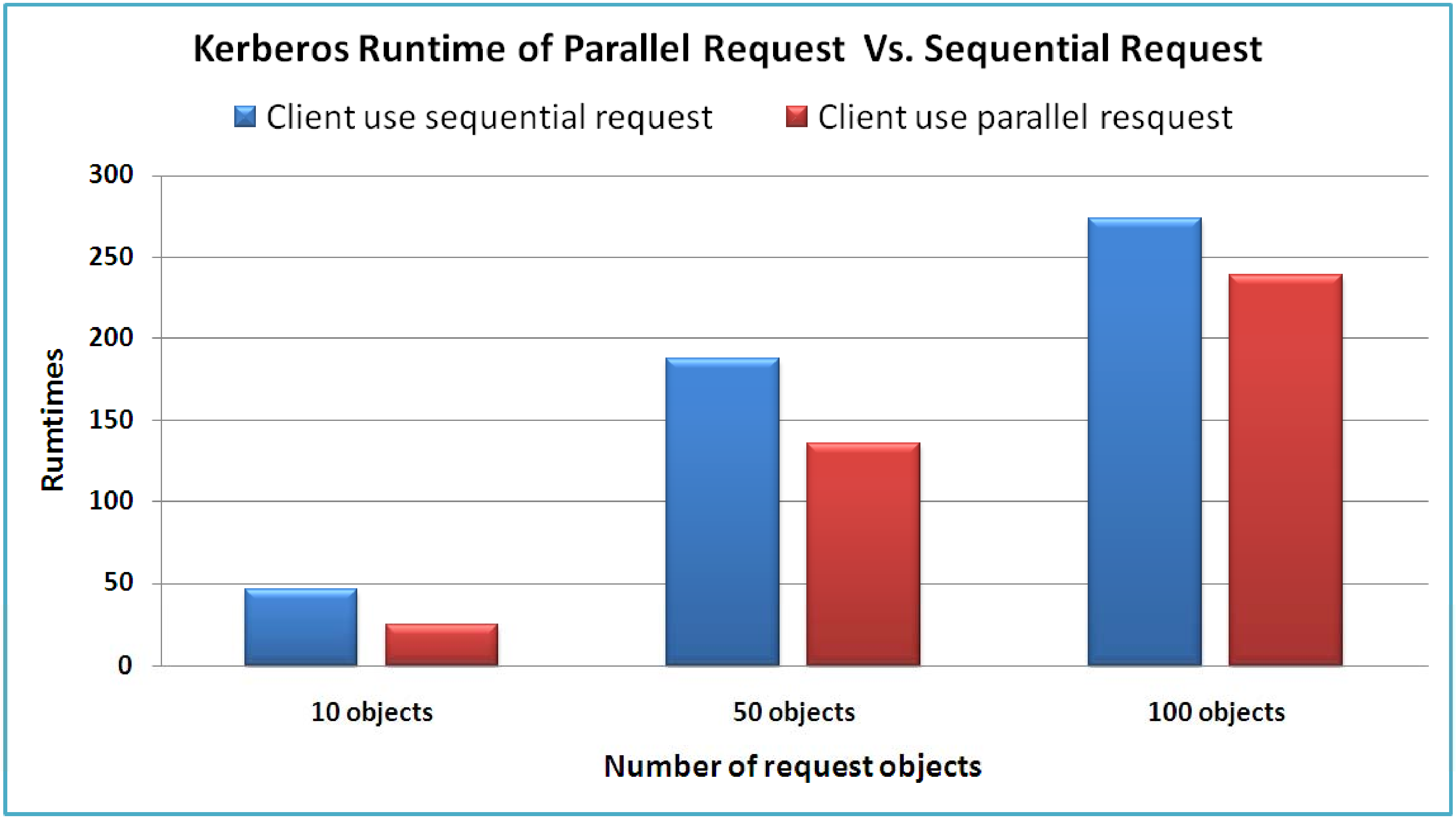
Performance of Secure Web Service with WSE Technology

Performance Evaluation



Performance of WCF Vs ASMX Web Service Technology

Performance Evaluation



Improve performance of web service with parallel processing

Conclusion

- Implementation of Secure Web service using X.509, Kerberos, STS.
- Performance evaluation of Message level security for web services.
- Scalability issues arise when message size is huge on message level security.
- Comparison on WCF implementation over ASMX implementation proved that WCF gives more throughput than ASMX.